

Министерство сельского хозяйства Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ
ИМЕНИ ИМПЕРАТОРА ПЕТРА I»

УТВЕРЖДАЮ

Декан экономического факультета

 А.Н. Черных

«21» мая 2024г.



РАБОЧАЯ ПРОГРАММА ПО ДИСЦИПЛИНЕ

Б1.В.15 Языки программирования

Направление: 09.03.03 Прикладная информатика

Профиль: Информационные системы и технологии в менеджменте АПК

Квалификация выпускника: бакалавр

Факультет Экономический

Кафедра Информационного обеспечения и моделирования агроэкономических систем

Разработчик рабочей программы:

Должность:

Ученая степень:

Ученое звание:

Кононова Наталья Николаевна

доцент

кандидат экономических наук

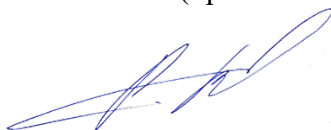


Воронеж-2024

Рабочая программа разработана в соответствии с Федеральным государственным образовательным стандартом высшего образования по направлению подготовки 09.03.03 Прикладная информатика (уровень бакалавриата) (утвержден приказом Министерства образования и науки РФ от 19 сентября 2017 № 922).

Рабочая программа утверждена на заседании кафедры Информационного обеспечения и моделирования агроэкономических систем (протокол № 8 от 26.04.2024 г.)

Заведующий кафедрой:



Р.В. Подколзин

Рабочая программа рекомендована к использованию в учебном процессе на заседании методической комиссии экономического факультета (протокол №9 от 21.05.2024 г.)

Председатель методической комиссии:



Л.В. Брянцева

Рецензент: начальник отдела информационно-коммуникационных технологий ООО "Овощ-Прод-Холдинг" А.П. Сухоедов

Содержание рабочей программы

1. Общая характеристика дисциплины
 - 1.1. Цель дисциплины
 - 1.2. Задачи дисциплины
 - 1.3. Предмет дисциплины
 - 1.4. Место в образовательной программе
 - 1.5. Связь с другими дисциплинами
 - 1.6. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья
2. Планируемые результаты изучения дисциплины
3. Объем дисциплины и виды учебной работы
 - 3.1. Очная форма обучения
 - 3.2. Заочная форма обучения
4. Содержание дисциплины
 - 4.1. Содержание дисциплины в разрезе разделов и подразделов
 - 4.2. Распределение контактной и самостоятельной работы по подразделам
 - 4.3. Перечень тем и учебно-методического обеспечения для самостоятельной работы обучающихся
5. Фонд оценочных средств
 - 5.1. Этапы формирования компетенций
 - 5.2. Шкалы и критерии оценивания достижения компетенций
 - 5.2.1. Шкалы академических оценок освоения дисциплины
 - 5.2.2. Критерии оценки достижения компетенций в ходе освоения дисциплины
 - 5.3. Материалы для оценки достижения компетенций
 - 5.3.1. Вопросы к экзамену
 - 5.3.2. Задания к экзамену
 - 5.3.3. Вопросы к зачету с оценкой
 - 5.3.4. Вопросы к зачету
 - 5.3.5. Темы курсового проекта (работы) и вопросы к защите
 - 5.3.4.1. Темы курсового проекта (работы)
 - 5.3.4.2. Вопросы к защите курсового проекта (работы)
 - 5.3.6. Вопросы тестов
 - 5.3.7. Вопросы для устного опроса
 - 5.3.8. Задания для проверки формирования умений и навыков
 - 5.4. Система оценивания достижения компетенций
 - 5.4.1. Оценка достижения компетенций в ходе промежуточной аттестации
 - 5.4.2. Оценка достижения компетенций в ходе текущего контроля
6. Учебно-методическое и информационное обеспечение дисциплины
 - 6.1. Рекомендуемая литература
 - 6.2. Ресурсы сети Интернет
 - 6.2.1. Электронные библиотечные системы
 - 6.2.2. Профессиональные базы данных и информационные системы
 - 6.2.3. Сайты и информационные порталы
7. Материально-техническое и программное обеспечение дисциплины
 - 7.1. Помещения для ведения образовательного процесса и оборудование
 - 7.2. Программное обеспечение
 - 7.2.1. Программное обеспечение общего назначения
 - 7.2.2. Специализированное программное обеспечение
8. Междисциплинарные связи

1. Общая характеристика дисциплины

1.1. Цель дисциплины:

формирование знаний, умений и навыков работы с различными средствами программирования и отладки для создания программного обеспечения на языках высокого уровня; обучение приемам разработки компьютерных приложений в профессиональной деятельности

1.2. Задачи дисциплины:

овладение основами языка, изучение переменных, применение циклов и ветвления;

изучение классов, использование коллекций, работа с методами;

овладение навыками отладки приложений, формирование исполняемого файла;

изучение работы с Visual Studio;

изучение основ разработки оконных приложений для ПК;

овладение основами языка 1с, формирование знаний, умений для работы с данными;

формирование знаний, умений и навыков для решения прикладных задач.

1.3. Предмет дисциплины:

методы и инструменты разработки приложений на языках программирования

1.4. Место в образовательной программе:

часть, формируемая участниками образовательных отношений

1.5. Взаимосвязь с другими дисциплинами:

Б1.О.15 Алгоритмизация и программирование

Б1.О.25 Проектный практикум

Б1.В.16 Разработка приложений для мобильных устройств

Б1.В.17 Программирование офисных приложений

1.6. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

определяются в индивидуальном порядке исходя из специфики заболевания и требований, указанных в Основной образовательной программе

2. Планируемые результаты изучения дисциплины

Компетенция		Индикатор достижения компетенции	
Код	Содержание	Код	Содержание
ПК-02	Способность разрабатывать и адаптировать прикладное программное обеспечение	32	лексические, синтаксические и семантические правила языка программирования
		33	функциональные возможности систем программирования
		У2	использовать языки программирования для решения профессиональных задач
		У3	разрабатывать программы с использованием систем программирования
		Н2	разработки программ на языках программирования высокого уровня
		Н3	использования систем программирования при решении профессиональных задач

3. Объем дисциплины и виды учебной работы

3.1. Очная форма обучения

Показатели	Семестр			Всего
	4	5	6	
Общая трудоёмкость, з.е./ч	4 / 144	4 / 144	4 / 144	12 / 432
Общая контактная работа, ч	58,15	42,15	50,75	151,05
Общая самостоятельная работа, ч	85,85	101,85	93,25	280,95
Контактная работа при проведении учебных занятий, в т.ч. (ч)	58,00	42,00	50,00	150,00
лекции	20	14	16	50,00
практические-всего	38	28	34	100,00
Самостоятельная работа при проведении учебных занятий, ч	77,00	93,00	75,50	245,50
Контактная работа при проведении промежуточной аттестации обучающихся, в т.ч. (ч)	0,15	0,15	0,75	1,05
групповые консультации	-	-	0,50	0,50
зачет	0,15	0,15	-	0,30
экзамен	-	-	0,25	0,25
Самостоятельная работа при промежуточной аттестации, в т.ч. (ч)	8,85	8,85	17,75	35,45
подготовка к зачету	8,85	8,85	-	17,70
подготовка к экзамену	-	-	17,75	17,75
Форма промежуточной аттестации	зачет	зачет	экзамен	зачет, экзамен

3. Объем дисциплины и виды учебной работы

3.2. Заочная форма обучения

Показатели	Курс		Всего
	2	3	
Общая трудоёмкость, з.е./ч	3 / 108	9 / 324	12 / 432
Общая контактная работа, ч	12,15	28,90	41,05
Общая самостоятельная работа, ч	95,85	295,10	390,95
Контактная работа при проведении учебных занятий, в т.ч. (ч)	12,00	28,00	40,00
лекции	6	14	20,00
практические-всего	6	14	20,00
Самостоятельная работа при проведении учебных занятий, ч	87,00	268,50	355,50
Контактная работа при проведении промежуточной аттестации обучающихся, в т.ч. (ч)	0,15	0,90	1,05
групповые консультации	-	0,50	0,50
зачет	0,15	0,15	0,30
экзамен	-	0,25	0,25
Самостоятельная работа при промежуточной аттестации, в т.ч. (ч)	8,85	26,60	35,45
подготовка к зачету	8,85	8,85	17,70
подготовка к экзамену	-	17,75	17,75
Форма промежуточной аттестации	зачет	зачет, экзамен	зачет, экзамен

4. Содержание дисциплины

4.1. Содержание дисциплины в разрезе разделов и подразделов

Раздел 1.

Основы языка программирования C#

Подраздел 1.1.

Основы языка, переменные, ветвление, циклы

Структура программы на языке C#, алфавит языка, идентификаторы, константы, выражения, операции; простейшие операторы ввода-вывода, форматы ввода и вывода; стандартные типы данных, описание типов данных, стандартные операции над типами данных; представление основных структур программирования

Подраздел 1.2.

Классы, коллекции, методы

Понятие класса и его структуры; встроенные коллекции: массивы, списки; объявление и вызов методов, передача параметров.

Подраздел 1.3.

Ошибки, отладка, компиляция кода

Обзор основных ошибок и обработка исключений; отладка приложений: точки останова, просмотр промежуточных значений переменных; формирование исполняемого файла

Раздел 2.

Создание оконных приложений

Подраздел 2.1.

Работа с Visual Studio

Основы работы в Visual Studio: создание проекта, компоновка и отладка приложений

Подраздел 2.2.

Основы разработки оконных приложений для ПК

Создание приложений, стандартные элементы управления и компоненты; работа с формами, визуализация данных.

Раздел 3.

Основы языка программирования 1С

Подраздел 3.1.

Основы языка 1С, работа с данными

Основные правила языка 1С, переменные, типы данных, условные команды и логические выражения; работа с массивами и циклами.

Подраздел 3.2.

Прикладной аспект работы с 1С

Системные процедуры и функции: работа с математическими и строковыми функциями, работа с датой и временем, специальные процедуры и функции, манипуляции с типами данных; язык запросов: формат текста описания запроса, константы и переменные запросов, атрибуты, доступные при описании внутренних переменных, операторы языка запросов; работа с файлами

4.2. Распределение контактной и самостоятельной работы по подразделам
Очная форма обучения

Разделы, подразделы дисциплины	Контактная работа		СР
	лекции	ПЗ	
Основы языка программирования С#			
Основы языка, переменные, ветвление, циклы	7,5	15,0	24,6
Классы, коллекции, методы	5,0	10,0	36,8
Ошибки, отладка, компиляция кода	5,0	10,0	36,8
Создание оконных приложений			
Работа с Visual Studio	7,5	15,0	36,8
Основы разработки оконных приложений для ПК	10,0	20,0	36,8
Основы языка программирования 1С			
Основы языка 1С, работа с данными	7,5	15,0	36,8
Прикладной аспект работы с 1С	7,5	15,0	36,8

**4.2. Распределение контактной и самостоятельной работы по подразделам
Заочная форма обучения**

Разделы, подразделы дисциплины	Контактная работа		СР
	лекции	ПЗ	
Основы языка программирования С#			
Основы языка, переменные, ветвление, циклы	3,0	3,0	35,6
Классы, коллекции, методы	2,0	2,0	53,3
Ошибки, отладка, компиляция кода	2,0	2,0	53,3
Создание оконных приложений			
Работа с Visual Studio	3,0	3,0	53,3
Основы разработки оконных приложений для ПК	4,0	4,0	53,3
Основы языка программирования 1С			
Основы языка 1С, работа с данными	3,0	3,0	53,3
Прикладной аспект работы с 1С	3,0	3,0	53,3

4.3. Перечень тем и учебно-методического обеспечения для самостоятельной работы обучающихся

Разделы, подразделы дисциплины	Учебно-методическое обеспечение	Объем часов СР	
		очная	заочная
Основы языка программирования C#			
Основы языка, переменные, ветвление, циклы	Бедердинова О. И. Программирование на языках высокого уровня [электронный ресурс]: Учебное пособие / О. И. Бедердинова, Т. А. Минеева - Москва: ООО "Научно-издательский центр ИНФРА-М", 2019 - 159 с. [ЭИ] [ЭБС Знаниум] URL: http://znanium.com/catalog/document?id=344897	24,6	35,6
Классы, коллекции, методы	Гуриков С. Р. Введение в программирование на языке Visual C# [электронный ресурс]: Учебное пособие / С. Р. Гуриков - Москва: Издательство "ФОРУМ", 2020 - 447 с. [ЭИ] [ЭБС Знаниум] URL: http://znanium.com/catalog/document?id=338986	36,8	53,3
Ошибки, отладка, компиляция кода	Хорев, П. Б. Объектно-ориентированное программирование с примерами на C# : учебное пособие / П.Б. Хорев. — Москва : ФОРУМ : ИНФРА-М, 2023. — 200 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-00091-680-3. - Текст : электронный. - URL: https://znanium.com/catalog/product/1926392	36,8	53,3
Создание оконных приложений			
Работа с Visual Studio	Гуриков С. Р. Введение в программирование на языке Visual C# [электронный ресурс]: Учебное пособие / С. Р. Гуриков - Москва: Издательство "ФОРУМ", 2020 - 447 с. [ЭИ] [ЭБС Знаниум] URL: http://znanium.com/catalog/document?id=338986	36,8	53,3
Основы разработки оконных приложений для ПК	Копырин, А. С. Программирование на C# в Visual Studio 2013 : учебное пособие / А. С. Копырин, Т. Л. Салова. - Москва : ФЛИНТА, 2021. - 54 с. - ISBN 978-5-9765-4754-4. - Текст : электронный. - URL: https://znanium.com/catalog/product/1851994	36,8	53,3
Основы языка программирования 1С			
Основы языка 1С, работа с данными	Бедердинова О. И. Программирование на языках высокого уровня [электронный ресурс]: Учебное пособие / О. И. Бедердинова, Т. А. Минеева - Москва: ООО "Научно-издательский центр ИНФРА-М", 2019 - 159 с. [ЭИ] [ЭБС Знаниум] URL: http://znanium.com/catalog/document?id=344897	36,8	53,3

Прикладной аспект работы с 1С	Бедердинова О. И. Программирование на языках высокого уровня [электронный ресурс]: Учебное пособие / О. И. Бедердинова, Т. А. Минеева - Москва: ООО "Научно-издательский центр ИНФРА-М", 2019 - 159 с. [ЭИ] [ЭБС Знаниум] URL: http://znanium.com/catalog/document?id=344897	36,8	53,3
Итого		245,5	355,5

5. Фонд оценочных средств

5.1. Этапы формирования компетенций

Разделы, подразделы дисциплины	Компетенции и ИД
	ПК-02
Основы языка программирования C#	
Основы языка, переменные, ветвление, циклы	32, 33, У2, У3, Н2, Н3
Классы, коллекции, методы	32, 33, У2, У3, Н2, Н3
Ошибки, отладка, компиляция кода	32, 33, У2, У3, Н2, Н3
Создание оконных приложений	
Работа с Visual Studio	32, 33, У2, У3, Н2, Н3
Основы разработки оконных приложений для ПК	32, 33, У2, У3, Н2, Н3
Основы языка программирования 1С	
Основы языка 1С, работа с данными	32, 33, У2, У3, Н2, Н3
Прикладной аспект работы с 1С	32, 33, У2, У3, Н2, Н3

5.2. Шкалы и критерии оценивания достижения компетенций

5.2.1. Шкалы академических оценок освоения дисциплины

Вид оценки	Оценки			
Академическая оценка по 4-х балльной шкале	неудовлетворительно	удовлетворительно	хорошо	отлично

Вид оценки	Оценки	
Академическая оценка по 2-х балльной шкале	не зачетно	зачтено

5.2.2. Критерии достижения компетенций в ходе освоения дисциплины

Критерии оценки на экзамене

Оценка, уровень достижения	Описание критериев
Отлично, высокий	Студент показал полные и глубокие знания материала, логично и аргументировано ответил на все вопросы экзаменационного билета, а также на дополнительные вопросы, способен самостоятельно решать сложные задачи дисциплины
Хорошо, продвинутый	Студент твердо знает материал, грамотно его излагает, не допускает существенных неточностей в ответе, достаточно полно ответил на вопросы экзаменационного билета и дополнительные вопросы, способен самостоятельно решать стандартные задачи дисциплины
Удовлетворительно, пороговый	Студент показал знание только основ материала, усвоил его поверхностно, но не допускал грубых ошибок или неточностей, требует наводящих вопросов для правильного ответа, не ответил на дополнительные вопросы, способен решать стандартные задачи дисциплины с помощью преподавателя
Неудовлетворительно, компетенции не освоены	Студент не знает основ материала, допускает грубые ошибки в ответе, не способен решать стандартные задачи дисциплины даже с помощью преподавателя

Критерии оценки на зачете

Оценка, уровень достижения	Описание критериев
Зачтено, высокий	Студент выполнил все задания, предусмотренные программой, отчитался об их выполнении, демонстрируя отличное знание освоенного материала и умение самостоятельно решать сложные задачи дисциплины
Зачтено, продвинутый	Студент выполнил все задания, предусмотренные программой, отчитался об их выполнении, демонстрируя хорошее знание освоенного материала и умение самостоятельно решать стандартные задачи дисциплины
Зачтено, пороговый	Студент выполнил все задания, предусмотренные программой, отчитался об их выполнении, демонстрируя знание основ освоенного материала и умение решать стандартные задачи дисциплины с помощью преподавателя
Не зачтено, компетенции не освоены	Студент выполнил не все задания, предусмотренные программой или не отчитался об их выполнении, не подтверждает знание освоенного материала и не умеет решать задачи дисциплины даже с помощью преподавателя

5.3. Материалы для оценки достижения компетенций

5.3.1. Вопросы к экзамену

№	Содержание	Компетенция	ИД
1	Активизация ИСР и выход из среды.	ПК-02	32
2	Окна ИСР.	ПК-02	32
3	Окно Конструктора формы.	ПК-02	32
4	Окно Редактора кода.	ПК-02	32
5	Окно инструментов.	ПК-02	32
6	Отличия проектов Приложение и Консольное приложение.	ПК-02	32
7	Создание консольного приложения.	ПК-02	32
8	Классы и ООП.	ПК-02	32
9	Что такое виртуальная машина в C#?	ПК-02	32
10	Определение класса.	ПК-02	32
11	Методы класса. Поля класса	ПК-02	32
12	Интерфейс.	ПК-02	32
13	Множественное наследование с помощью интерфейсов.	ПК-02	32
14	Алфавит языка, операции, идентификаторы.	ПК-02	32
15	Структура программы.	ПК-02	32
16	Переменные. Их объявление.	ПК-02	32
17	Форматы представления чисел (с фиксированной и плавающей точкой).	ПК-02	32
18	Типы целых чисел без знака: Byte.	ПК-02	32
19	Типы целых чисел со знаком: int, Int16, Int32, Int64, sbyte.	ПК-02	32
20	Типы вещественных чисел(с плавающей точкой): double Single.	ПК-02	32
21	Тип чисел decimal.	ПК-02	32
22	Как организовать разветвление вычислений?	ПК-02	32
23	Ветвление if...else. Вложенные ветвления.	ПК-02	33
24	Инструкция выбора switch. Фраза case.	ПК-02	33
25	Зачем во фразе case применяется оператор break?	ПК-02	33
26	Отличия цикла с заданным числом повторений от цикла с неизвестным числом повторений.	ПК-02	32
27	Синтаксис инструкции while.	ПК-02	33
29	Синтаксис инструкции do...while.	ПК-02	32
30	Синтаксис инструкции foreach.	ПК-02	32
31	Вычисление суммы членов бесконечного ряда по рекуррентной формуле.	ПК-02	32
32	Условие выхода из цикла при вычислении суммы членов бесконечного ряда.	ПК-02	32
33	Инструкция цикла for.	ПК-02	33
34	Как организовать цикл с нарастанием индекса?	ПК-02	32
35	Как организовать цикл с убыванием индекса?	ПК-02	32
36	Организация вычисления суммы и произведения.	ПК-02	32
37	Строки. Массивы и файлы.	ПК-02	32
38	Какие операторы языка можно использовать для описания массивов?	ПК-02	32
39	Особенности организации цикла при обработке массивов?	ПК-02	32
40	Особенность ввода и вывода массивов?	ПК-02	У2
41	Представление строковых переменных типа String как одномерных массивов.	ПК-02	У2
42	Вложенные массивы.	ПК-02	У2
43	Типы файлов, определенные в языке	ПК-02	У2
44	Способы связи с файлами.	ПК-02	У2
45	Процедуры создания, открытия, закрытия файлов.	ПК-02	32
46	Обработка ошибок и исключений.	ПК-02	32
47	Классы исключений и их свойства.	ПК-02	32
48	Инструкция try catch. Назначение структура и применение.	ПК-02	32
49	Исключение EZeroDivide для целых чисел.	ПК-02	33
50	Исключение EZeroDivide для вещественных чисел.	ПК-02	33
51	Исключение ERRangeError.	ПК-02	33

5.3.2. Задания к экзамену

№	Содержание	Компетенция	ИД
1	Написать фрагмент программного кода , используя различные типы данных	ПК-02	Н3
2	Написать фрагмент программного кода , используя массивы	ПК-02	У3
3	Написать фрагмент программного кода , используя списки	ПК-02	У3
4	Написать фрагмент программного кода , используя операции checked и unchecked	ПК-02	Н2
5	Написать фрагмент программного кода , используя операции is, sizeof	ПК-02	Н2
6	Написать фрагмент программного кода , используя операции as, typeof	ПК-02	Н2
7	Написать фрагмент программного кода , используя операции is, sizeof	ПК-02	Н2
8	Написать фрагмент программного кода , используя различные классы	ПК-02	У3
9	Написать фрагмент программного кода , используя объявления и вызов методов	ПК-02	У3
10	Написать фрагмент программного кода , используя параметр ref	ПК-02	Н2
11	Написать фрагмент программного кода , используя параметр out	ПК-02	Н2
12	Написать фрагмент программного кода , используя поле readonly	ПК-02	Н2
13	Написать фрагмент программного кода , используя элементы управления Button, CheckBox	ПК-02	Н2
14	Написать фрагмент программного кода , используя элементы управления RadioButton, ComboBox	ПК-02	Н2
15	Написать фрагмент программного кода , используя элементы управления DateTimePicker, ImageList	ПК-02	Н2
16	Написать фрагмент программного кода , используя элементы управления ListView, TextBox	ПК-02	Н2
17	Написать фрагмент программного кода , используя элементы управления TabControl, TabPage	ПК-02	Н2

5.3.3. Вопросы к зачету с оценкой

Не предусмотрено

5.3.4. Вопросы к зачету

№	Содержание	Компетенция	ИД
1	История языка C# и ИСР.	ПК-02	32
2	Назначение и содержание главного окна ИСР.	ПК-02	32
3	Главное меню ИСР. Опции пунктов.	ПК-02	32
4	Использование встроенного подсказчика.	ПК-02	32
5	Создание Windows приложения.	ПК-02	32
6	Объектно-ориентированное программирование.	ПК-02	32
7	Поля класса.	ПК-02	32
8	Типы данных.	ПК-02	32
9	Совместимость типов при присвоениях.	ПК-02	32
10	Какие структуры вычислительных процессов Вы знаете?	ПК-02	32
11	Циклический процесс с неизвестным числом повторений.	ПК-02	32
12	Инструкции языка для организации циклов. Их сравнение.	ПК-02	32
13	Прямое вычисление суммы членов бесконечного ряда.	ПК-02	32
14	Преимущества использования инструкций циклов в программе.	ПК-02	32
15	Что такое массив?	ПК-02	32
16	Описание типа -массив.	ПК-02	32
17	Особенности программирования при обработке массивов?	ПК-02	У2
18	Файловая переменная.	ПК-02	32

5.3.5. Темы курсового проект (работы) и вопросы к защите

Не предусмотрено

5.3. Материалы для оценки достижения компетенций

5.3.6. Вопросы тестов

№	Содержание	Компетенция	ИД
1—это поименованная или иным образом идентифицированная часть компьютерной программы, содержащая описание определённого набора действий, которая может быть многократно вызвана из разных частей программы.	ПК-02	32
2	Что будет результатом выполнения следующей функции при n=4? function f(n:word): longint; begin if (n=1) or (n=0) then f:=1 else f:=n*f(n-1); end;	ПК-02	33
3	В какой части программы описываются подпрограммы ? (?) Между заголовком и разделом описания (?) В разделе описания до программного блока (?) В программном блоке после begin	ПК-02	33
4	Какой оператор должен обязательно присутствовать в операторном блоке функции? (?) Любой оператор, использующий ее входные параметры (?) Условный оператор (?) Оператор цикла (?) Оператор присваивания имени функции, результата работы подпрограммы	ПК-02	32
5	Как организуется передача выходных параметров(результатов работы) из подпрограммы процедура в основную программу ? (?) При помощи параметров переменных, с использованием служебного слова Var, например, var x: real (?) При помощи параметров значений, например, x: real (?) При помощи параметров констант, с использованием служебного слова const, например, const x: real	ПК-02	32
6	Что пропущено в данной записи процедуры? procedure test(..... var t: integer); begin t:=x+c; end; (?) выходные параметры (?) промежуточные переменные (?) название процедуры (?) входные параметры	ПК-02	33
7	В какой строке программы допущена ошибка? 1 Var a, b : longint; 2 function NOD (a,b:longint):longint; 3 begin 4 if a>b then NOD := NOD(a-b,b) 5 else if a<b then NOD := NOD (a,b-a) 6 else NOD := a; 7 end; 8 begin {основная программа} 9 readln (a,b); 10 NOD (a,b,n); 11 writeln (a-b); 12 end.	ПК-02	33
8	В приведенном списке выберите две процедуры: (?) Read (?) Sqr (?) Random (?) Readln	ПК-02	32

9	Определить результат работы следующего фрагмента программы k:=6; for i:=1 to 5 do inc(k); write(k, ' ');	ПК-02	32
10	В результате выполнения последовательности операторов x:=sqr(2); y:=sqrt(x)-abs(-2); write(y) на экране будет выведено:	ПК-02	33
11	При выполнении последовательности операторов A:=4; B:=2; X:=A mod B значение переменной X равно:	ПК-02	33
12	Какие объекты можно использовать в качестве формальных параметров в процедуре? (?) константы, переменные, выражения (?) переменные с указанием их типа (?) выражения, константы (?) типизированные константы	ПК-02	32
13	Какая строка соответствует описанию связи файла с файловой переменной? (?) Add (F, C:\pascal\file.txt); (?) File Open (F, C:\pascal\file.dat); (?) Append (F, 'C:\pascal\file.doc'); (?) File (F, C:\pascal\file.pas&apos); (?) Assign (F, 'C:\pascal\file.dat');	ПК-02	33
14	Команда Rewrite (F)- открывает файл для	ПК-02	32
15	Команда Reset (F)- открывает файл для	ПК-02	32
16	Для чего используется работа с файлами ? (?) Для облегчения работы ЭВМ (?) Так принято в Pascal (e) (?) Для программирования игр (?) Для освобождения оперативной памяти (?) Для записи результатов в файл, и чтения данных из файлов	ПК-02	32
17	Имеется описание следующего вида type date = record month: 1..12; day: 1..31; year: integer end; var d: date; Укажите правильное обращение к полю ДЕНЬ (?) d.day:=12; (?) day=12; (?) day.d:=12;	ПК-02	33
18	Современная система программирования, в основе которой лежит объектно-ориентированный язык Pascal и визуальные средства программирования, называется (?) Pascal (?) Basic (?) Delphi (?) Assembler	ПК-02	32
19	К базовым принципам объектно-ориентированного подхода программирования относят (три правильных ответа): (?) полиморфизм (?) инкапсуляция (?) наследование (?) деструктуризация (?) глобализация	ПК-02	32
20	Процесс, посредством которого один объект может приобретать свойства другого, называется.....	ПК-02	32
21	Свойство, которое позволяет одно и то же имя использовать для решения двух или более схожих, но технически различных задач, называется	ПК-02	32
22	Множество объектов, обладающих одинаковыми свойствами, операциями, отношениями и семантикой, в объектно-ориентированной методологии называется.....:	ПК-02	32

23	Методы, разрабатываемые с учетом специфики решаемых задач, называются: (?) ориентированными на область применения (?) специфическими (?) прикладными	ПК-02	32
24	Что может называться компонентами ? (?) Форма (?) Кнопки (?) Свойства объектов (?) Надписи	ПК-02	32
25	Форма, из которой вызываются все остальные зависимые формы, называется.....	ПК-02	32
26	К какому классу средств разработки относится система C#: (?) Ускоренной разработки (?) Проектирования (?) Профессионального программирования (?) Мультимедиа	ПК-02	32
27	Макет, пробная версия программы, с целью проверки пригодности предлагаемых для применения концепций, архитектурных и/или технологических решений, называется: (?) прототипом (?) релизом (?) образцом	ПК-02	32
28	Свойства программного обеспечения, которые должны быть надлежащим образом представлены в нём для решения конкретных практических задач, называются: (?) требованиями (?) условиями программирования (?) данные программы	ПК-02	32
29 –это результат процессов программирования и проектирования, который объединяет программный код и графический интерфейс, а так же включает в себя программные модули форм и самостоятельные программные модули в виде отдельных файлов.	ПК-02	32
30	Команда Главного меню- Run, выполняет (какое действие)..... проекта на выполнение:	ПК-02	33
31	Функция обработки строк- IntToStr (S), преобразует число в строку:	ПК-02	32
32	Функция обработки строк- StrToFloat(S) , преобразует строку вчисло:	ПК-02	32
33	Команда «Save all» необходима для всех файлов проекта	ПК-02	32
34	Компонент (два правильных ответа), предназначенный для размещения на форме текстовой информации: (?) Edit (?) Label (?) Memo (?) ListBox	ПК-02	32
35	Компонент, представляющий собой однострочное редактируемое текстовое поле, называется..... :	ПК-02	32
36	Компонент, представляющий собой поле для редактирования, который дает возможность ввести данные путем набора с клавиатуры или выбором из предложенного списка, называется	ПК-02	33
37	Button – это компонент, представляющий командную	ПК-02	32
38	Компонент, представляет собой переключатель, который может быть находиться в одном из двух состояний: (?) CheckBox (?) RadioButton (?) Edit (?) Button	ПК-02	32
39	Компонент, представляющий собой комбинацию списка и однострочного текстового поля: (?) ComboBox (?) StringGrid (?) Form (?) ListBox	ПК-02	32

40	<p>Многостраничные панели позволяет создавать компонент:</p> <p>(?) ComboBox (?) StringGrid (?) Form (?) TPageControl</p>	ПК-02	32
41	<p>Компонент необходимый для визуального оформления формы:</p> <p>(?) CheckBox (?) RadioButton (?) GroupBox (?) Button</p>	ПК-02	32
42	Enabled -это признак компонента	ПК-02	32
43	<p>Доступ к базе данных в Delphi осуществляют компоненты (два правильных ответа):</p> <p>(?) ADOConnection (?) ADODataset (?) ADOTable</p>	ПК-02	32
44	<p>В каком свойстве компонента ADODatasete прописывается SQL- команда?</p> <p>(?) Connection (?) CommandText (?) Activate (?) DataSet</p>	ПК-02	32
45	<p>Какой компонент обеспечивает перемещение указателя текущей записи к следующей, предыдущей, первой или последней записи, а также выполнение других операций в результате щелчка на соответствующей кнопке?</p> <p>(?) DBEdit (?) DBNavigator (?) DDDBMemo (?) DBGrid</p>	ПК-02	32
46	Компонент TDBGrid –обеспечивает отображение данных в режиме	ПК-02	32
47	<p>Отображение данных в режиме формы обеспечивает компонент</p> <p>(?) DBEdit (?) TEdit (?) TDBGrid (?) TLabel</p>	ПК-02	32
48	<p>Какой минимальный набор команд должен находиться в SQL- запросе, обеспечивающий выборку данных из базы данных (таблицы)</p> <p>(?) SELECT СписокПолей FROM Таблица WHERE (Критерий) (?) SELECT СписокПолей FROM Таблица WHERE (Критерий) ORDER BY СписокПолей (?) DISTINCT список выбора FROM имя таблицы GROUP BY имя столбца (?) SELECT СписокПолей FROM Таблица</p>	ПК-02	32
49	<p>Какой оператор при использовании шаблонов в запросах среды Delphi вместо оператора = необходимо использовать</p> <p>(?) == (?) := (?) Like (?) WHERE</p>	ПК-02	32
50	<p>Использование какого оператора в SQL- запросах позволяет получить набор данных из базы данных без повторяющихся значений?</p> <p>(?) Distinct (?) From (?) Order By (?) Group By</p>	ПК-02	32
51	Group By –это оператор, после какого указывается список полей, по которым производится	ПК-02	32
52	Агрегирующая функция Avg в SQL- запросах возвращает величину (различных) значений?	ПК-02	32
53	Агрегирующая функция Count в SQL- запросах возвращает (различных) значений?	ПК-02	32

54	С помощью какого ключевого слова в SQL- запросах можно изменить порядок сортировки по убыванию в каждом отдельном случае? (?) Desc (?) Asc (?) Order By (?) Group By	ПК-02	32
55	Left join-команда в SQL- запросах возвращает строки из таблицы, даже если их нет в правой таблице?	ПК-02	32
56	Right join- команда в SQL- запросах возвращает строки изтаблицы, даже если их нет в левой таблице?	ПК-02	32
57	Какая команда в SQL- запросах возвращает строки, когда есть хотя бы одно совпадение в обеих таблицах? (?) Left join (?) Right join (?) Inner join	ПК-02	32
58	Данный транслятор полностью обрабатывающий весь текст программы, прежде чем запустить ее на исполнение, называется	ПК-02	32
59	Данный транслятор, который берет очередной оператор языка из текста программы, анализирует его структуру и затем сразу исполняет, только после того как текущий оператор успешно выполнен, далее переходит к следующему, называется	ПК-02	32
60	Выявление и устранение неправильной работы программы называется.....:	ПК-02	33
61	Что такое подпрограмма? (?) алгоритм, имеющий циклическую структуру (?) именованная последовательность команд (?) алгоритм, имеющий разветвляющую структуру (?) любой участок кода	ПК-02	33
62	Какие существуют подпрограммы в языке Паскаль? (?) Процедуры (?) Методы (?) Функции (?) Свойства (?) Операторы	ПК-02	33
63	Локальные переменные - это (?) переменные, написанные строчными буквами (?) переменные, объявленные внутри подпрограмм и доступные только ей самой (?) переменные, имеющие тип LOCAL (?) фиктивно(формально) присутствующие в заголовке процедуры и определяющие тип и место постановки фактических параметров	ПК-02	32
64	Что будет результатом выполнения следующей функции при n=4? function f(n:word): longint; begin if (n=1) or (n=0) then f:=1 else f:=n*f(n-1); end; (?) 12 (?) 21 (?) 24 (?) 0	ПК-02	33
65	В какой части программы описываются подпрограммы в языке Паскаль? (?) Между заголовком и разделом описания (?) В разделе описания до программного блока (?) В программном блоке после begin	ПК-02	32
66	Какой оператор должен обязательно присутствовать в операторном блоке функции? (?)Любой оператор, использующий ее входные параметры (?) Условный оператор (?) Оператор цикла (?) Оператор присваивания имени функции, результата работы подпрограммы	ПК-02	32

67	<p>При помощи чего реализуются вспомогательные алгоритмы при программировании?</p> <p>(?) при помощи ветвлений (?) при помощи циклов (?) при помощи следований (?) при помощи подпрограмм</p>	ПК-02	32
68	<p>Как организуется передача входных параметром (исходных данных) в подпрограмму?</p> <p>(?) При помощи параметров переменных, с использованием служебного слова Var, например, var x: real (?) При помощи параметров значений, например, x: real (?) При помощи параметров констант, с использованием служебного слова const, например, const x: real</p>	ПК-02	32
69	<p>Как организуется передача выходных параметром(результатов работы) из подпрограммы процедура в основную программу ?</p> <p>(?) При помощи параметров переменных, с использованием служебного слова Var, например, var x: real (?) При помощи параметров значений, например, x: real (?) При помощи параметров констант, с использованием служебного слова const, например, const x: real (?) При помощи имени подпрограммы</p>	ПК-02	32
70	<p>Как организуется передача выходных параметром(результатов работы) из подпрограммы функция в основную программу ?</p> <p>(?) При помощи параметров переменных, с использованием служебного слова Var, например, var x: real (?) При помощи параметров значений, например, x: real (?) При помощи параметров констант, с использованием служебного слова const, например, const x: real (?) При помощи имени подпрограммы</p>	ПК-02	32
71	<p>Чем функции отличаются от процедур?</p> <p>(?) Функция принимает на вход несколько параметром, а процедура -один (?) Функция возвращает одно значение, а процедура может вернуть несколько (?) Функция возвращается в составе другого оператора или команды, а вызов процедуры- это отдельная команда (?) Результатом работы функция может быть значение только простого типа, а процедуры- любого (?) Функция может возвращать данные только логического типа, а процедура- численных</p>	ПК-02	32
72	<p>В приведенном заголовке процедуры укажите параметры - переменные: Procedure Sum (a,b: integer; Var c, d: integer;);</p> <p>(?) a,b (?) c,d (?)a,c (?)b,d</p>	ПК-02	32
73	<p>В приведенных фрагментах описания заголовков функций выберите правильные:</p> <p>(?) Function Sum1 (a,b:Integer; Var c:integer): real; (?) function Sum (a: integer; Var c: integer); (?) Function 2S (a, b): integer; (?) Function SQ (a,c: real): integer;</p>	ПК-02	У3
74	<p>В приведенном списке выберите функции:</p> <p>(?) Randomize (?) Random (?) Abs (?) Round</p>	ПК-02	32
75	<p>В приведенном списке выберите процедуры:</p> <p>(?) Read (?) Sqr (?) Random (?) Readln</p>	ПК-02	32

76	<p>В приведенном заголовке процедуры укажите параметры - значения: Procedure kl (a,b: Integer; c: Real; Var k: String);</p> <p>(?) a,b,c (?) c (?) k (?) a,b,c,k</p>	ПК-02	32
77	<p>Глобальными переменными называются</p> <p>(?) переменные, которые описаны в самой программе, а используются только в процедуре</p> <p>(?) переменные, которые описаны в самой программе, а используются как в программе, так и в процедуре</p> <p>(?) переменные, которые описаны в процедуре и используются только в процедуре</p> <p>(?) переменные, которые описаны в процедуре, а используются в программе</p>	ПК-02	32
78	<p>Формальными параметрами называются:</p> <p>(?) переменные, описанные внутри подпрограммы и используемые только внутри подпрограммы</p> <p>(?) переменные, которые описаны в самой программе, а используются как в программе, так и в подпрограмме</p> <p>(?) переменные, описанные в заголовке подпрограммы и используемые только внутри подпрограммы</p>	ПК-02	32
79	Определите, какие переменные в приведенном фрагменте программы локальные.	ПК-02	32
80	Program Slon;	ПК-02	32
81	<p>Что такое рекурсия?</p> <p>(?) Объект, который частично определяется через другие неизвестные объекты, называется - рекурсивным.</p> <p>(?) Объект, который частично определяется через самого себя, называется - рекурсивным.</p> <p>(?) Объект, который частично определяется через другие известные объекты, называется - рекурсивным.</p>	ПК-02	32
82	<p>Укажите правильный формат заголовка процедуры:</p> <p>(?) Procedure ();</p> <p>(?) Procedure (): «тип»;</p> <p>(?) Procedure ();;</p> <p>(?) Procedure ():</p>	ПК-02	32
83	<p>В приведенных фрагментах описания заголовков процедур выберите ошибочные:</p> <p>(?) Procedure Sum1 (a,b:Integer; Var c:integer): real;</p> <p>(?) Procedure S2m3;</p> <p>(?) Procedure 2S (a: Real; b: integer);</p> <p>(?) Procedure SQ (a,c: array[1..10] of real);</p>	ПК-02	33
84	<p>Укажите правильный формат заголовка функции:</p> <p>(?) Function ();</p> <p>(?) Fанction ();</p> <p>(?) Function (): ;</p> <p>(?) Fанction (): ;</p>	ПК-02	32
85	<p>Каковы отличия функции от процедуры?</p> <p>(?) Процедура может иметь несколько значений, а функция только одно (оно и будет ее результатом). Кроме того, в теле функции обязательно должен быть хотя бы один оператор присвоения, где в левой части стоит имя функции, а в правой её значение. Иначе значение не будет определено.</p> <p>(?) Функция может иметь несколько значений, а процедура только одно (оно и будет ее результатом). Кроме того, в теле процедуры обязательно должен быть хотя бы один оператор присвоения, где в левой части стоит имя процедуры, а в правой её значение. Иначе значение не будет определено.</p> <p>(?) Процедура может иметь только одно значение, как и функция. Кроме того, в теле процедуры обязательно должен быть хотя бы один оператор присвоения, где в левой части стоит имя процедуры, а в правой - её значение.</p>	ПК-02	32

86	<p>Неверно записано описание переменных:</p> <p>(?) VAR A,B:real; I,J:integer;</p> <p>(?) VAR j,g,t:integer; i:real;</p> <p>(?) VAR i,max:integer; a,max:real;</p> <p>(?) VAR a,b:real; c,d:real;</p>	ПК-02	32
87	<p>Имеется описание заголовка процедуры: procedure proba(a, b, c:integer); При использовании процедуры с таким заголовком значения фактических параметров передаются ...</p> <p>(?) по ссылке</p> <p>(?) по значению</p> <p>(?) как константа</p> <p>(?) как процедура</p>	ПК-02	32
88	<p>Определить результат работы следующего фрагмента программы</p> <pre>k:=6; for i:=1 to 5 do inc(k); write(k, ' ');</pre> <p>(?) 7 8 9 10 11</p> <p>(?) 11</p> <p>(?) 6</p> <p>(?) 1 2 3 4 5</p>	ПК-02	33
89	<p>Какое выражение неверно?</p> <p>(?) Структура подпрограммы аналогична структуре основной программы</p> <p>(?) Подпрограмма предназначена для выполнения какой-то последовательности действий</p> <p>(?) В Паскале имеется две разновидности подпрограмм – процедуры и функции</p> <p>(?) В подпрограмме нельзя использовать глобальные переменные</p> <p>Определить результат работы следующего фрагмента программы</p> <pre>k:=6; for i:=1 to 5 do begin inc(k); write(k, ' '); end;</pre> <p>(?) 7 8 9 10 11</p> <p>(?) 11</p> <p>(?) 6</p> <p>(?) 1 2 3 4 5</p>	ПК-02	33
90	<p>В Паскале различие в высоте букв (прописные или строчные)</p> <p>(?) имеет значение при написании текстовых констант</p> <p>(?) имеет значение для написания служебных слов</p> <p>(?) имеет значение для написания имен переменных</p> <p>(?) игнорируется</p>	ПК-02	32
91	<p>Какие виды подпрограмм есть в Паскале?</p> <p>(?) модуль и функция</p> <p>(?) процедура и функция</p> <p>(?) модуль и процедура</p> <p>(?) процедура</p>	ПК-02	33
92	<p>В результате выполнения последовательности операторов x:=sqr(2); y:=sqrt(4)-abs(-2); write(x, ' ', y) на экране будет выведено:</p> <p>(?) 0 -1</p> <p>(?) 4 6</p> <p>(?) 4 0</p> <p>(?) 1 0</p>	ПК-02	33
93	<p>При выполнении последовательности операторов A:=4; B:=2; X:=A mod B значение переменной X равно:</p> <p>(?) выражение записано неверно</p> <p>(?) 8</p> <p>(?) операция невыполнима, так как деление на 0 невозможно</p> <p>(?) 0</p>	ПК-02	33

94	<p>Подпрограммы располагаются в разделе описания...?</p> <p>(?) переменных (?) типов (?) процедур и функций (?) меток</p>	ПК-02	32
95	<p>Для вычисления экспоненты применяется процедура:</p> <p>(?) ORD(X) (?) SQR(X) (?) TRUNC(X) (?) EXP(X)</p>	ПК-02	32
96	<p>Для возведения в квадрат применяется функция:</p> <p>(?) LN(X) (?) LOG(X) (?) SQRT(X) (?) SQR(X)</p>	ПК-02	32
97	<p>Для вычисления квадратного корня применяется функция:</p> <p>(?) SQR(X) (?) SQRT(X) (?) ORD(X) (?) EXP(X)</p>	ПК-02	32
98	<p>Имеется описание заголовка процедуры: procedure proba(var a, b, c:integer); При использовании процедуры с таким заголовком значения фактических параметров передаются ...</p> <p>(?) по ссылке; (?) по значению. (?) как константа (?) как процедура</p>	ПК-02	33
99	<p>Имеется описание заголовка процедуры: procedure proba(a, b, c:integer); При использовании процедуры с таким заголовком значения фактических параметров передаются ...</p> <p>(?) по ссылке; (?) по значению (?) как константа (?) как процедура</p>	ПК-02	32
100	<p>Каких параметров подпрограммы не существует?</p> <p>(?) параметров-переменных (?) параметров-констант (?) параметров-значений (?) параметров-модулей</p>	ПК-02	32
101	<p>Где правильно описана процедура?</p> <p>(?) PROCEDURA (a:array [1..10] of integer; b:real) (?) PROSEDURA (var a:array [1..10] of integer; b:real) (?) PROSEDURA (a:array of integer; b:real) (?) PROCEDURE (a:array of integer; var b:real)</p>	ПК-02	32
102	<p>Какое служебное слово указывается в заголовке процедуры?</p> <p>(?) BEGIN (?) PROCEDURE (?) FUNCTION (?) PROCEDURA</p>	ПК-02	33
103	<p>Какие объекты можно использовать в качестве формальных параметров в процедуре?</p> <p>(?) константы, переменные, выражения (?) переменные с указанием их типа (?) выражения, константы (?) типизированные константы</p>	ПК-02	32

104	<p>Что представляют собой фактические параметры?</p> <p>(?) константы, переменные, выражения с указанием их типа</p> <p>(?) переменные с указанием их типа</p> <p>(?) выражения, константы</p> <p>(?) константы, переменные, выражения без указания типа</p>	ПК-02	32
105	<p>Каким символом разделяются перечень однотипных формальных параметров в процедуре?</p> <p>(?) запятой</p> <p>(?) точкой с запятой</p> <p>(?) двоеточием</p> <p>(?) точкой</p>	ПК-02	32
106	<p>Каким символом разделяются перечень разнотипных формальных параметров в процедуре?</p> <p>(?) запятой</p> <p>(?) точкой с запятой</p> <p>(?) двоеточием</p> <p>(?) точкой</p>	ПК-02	32
107	<p>Где правильно описана функция?</p> <p>(?) FUNCTION (a,b,c:real):integer;</p> <p>(?) FUNCTION Faktor (a:integer):integer;</p> <p>(?) FYNCTION Sum (a,b,c:real):integer;</p> <p>(?) FUNKTION A(a,b,c:real): integer;</p>	ПК-02	33
108	<p>Укажите правильное обращение к процедуре, объявленной как: Procedure Tangens(f:real; var t:real);</p> <p>(?) tangens (3.14,t,k);</p> <p>(?) tangens(pi/2,t);</p> <p>(?) tangens(a,3.14);</p> <p>(?) tangen(pi/6,t);</p>	ПК-02	33
109	<p>Укажите правильный вызов процедуры Procedure LOG10(Argument:Real; var LOG:real);</p> <p>(?) LOG10(x,2.3);</p> <p>(?) LOG10(x,2*y);</p> <p>(?) LOG10(1.5,y);</p> <p>(?) LOG10(x,y,z);</p>	ПК-02	33
110	<p>Длину строки можно определить с помощью процедуры:</p> <p>(?) LENGTH;</p> <p>(?) ORD;</p> <p>(?) CHR;</p> <p>(?) RECORD;</p>	ПК-02	32
111	<p>Переменная строкового типа обозначается идентификатором</p> <p>(?) string;</p> <p>(?) word;</p> <p>(?) char;</p> <p>(?) byte;</p>	ПК-02	32
112	<p>Строки программы в Турбо Паскале имеют максимальную длину:</p> <p>(?) 80 символов;</p> <p>(?) 126 символов;</p> <p>(?) 256 символов;</p> <p>(?) 512 символов.</p>	ПК-02	32
113	<p>Что появится на экране в результате выполнения следующих операторов: ws:='электрификация'; sw:='тр'; p:=pos(sw,ws); write(p:2);</p> <p>(?) 1</p> <p>(?) 3</p> <p>(?) и</p> <p>(?) 5</p>	ПК-02	32
114	<p>56. ORD(X) – функция, которая ...</p> <p>(?) возвращает порядковый номер значения выражения X</p> <p>(?) возвращает следующее значение порядкового типа</p> <p>(?) возвращает символ в верхнем регистре</p> <p>(?) преобразует выражение X в символ</p>	ПК-02	32

115	С текстовыми файлами могут работать файловые переменные, описанные следующим образом: (?) var f: text (?) var f: file of <тип_элементов> (?) var f: file !	ПК-02	32
116	С типизированными файлами могут работать файловые переменные, описанные следующим образом: (?) var f: text (?) var f: file of <тип_элементов> (?) var f: file !	ПК-02	33
117	С бинарными файлами могут работать файловые переменные, описанные следующим образом: (?) var f: text (?) var f: file of <тип_элементов> (?) var f: file !	ПК-02	33
118	Чтобы иметь возможность считывать информацию из бинарного файла, его нужно открыть при помощи команды (?) append(f) (?) close(f) (?) reset(f) (?) rewrite(f)	ПК-02	32
119	Чтобы иметь возможность записывать информацию в бинарный файл, его нужно открыть при помощи команды (?) append(f) (?) close(f) (?) rewrite(f)	ПК-02	32
120	Чтобы иметь возможность одновременно считывать информацию из текстового файла и записывать ее туда, файл нужно открыть при помощи команды (?) append(f) (?) close(f) (?) reset(f) (?) rewrite(f)	ПК-02	32
121	Какая последовательность символов будет содержаться в файле f: file of char после выполнения следующего отрывка программы: rewrite(f); for c:= 'a' to 'p' do write(f,c);{c: char} seek(f,10); read(f,c); write(f,c); seek(f,15); truncate(f); write(f,c); (?) abcdefghijklmnop (?) abcdefghijklmno (?) abcdefghijkkmnok (?) abcdefghijklmnopk	ПК-02	33

122	<p>Какая последовательность символов будет содержаться в файле f: file of char после выполнения следующего отрывка программы:</p> <pre>rewrite(f); for c:= 'a' to 'p' do write(f,c); {c: char} seek(f,10); read(f,c); truncate(f); seek(f,15); write(f,c);</pre> <p>(?) абвгдежзийклмноп (?) абвгдежзийклмнок (?) абвгдежзийлмнопрк (?) абвгдежзийклмнопк</p>	ПК-02	33
123	<p>Какая последовательность символов будет содержаться в файле f: file of char после выполнения следующего отрывка программы:</p> <pre>rewrite(f); for c:= '0' to '9' do write(f,c); {c: char} seek(f,5); read(f,c); write(f,c); seek(f,3); truncate(f); seek(f,9); write(f,c);</pre> <p>(?) 0123456785 (?) 0123455789 (?) 0123455785 (?) 0123456789</p>	ПК-02	33
124	<p>Какая строка должна быть при закрытии файла ?</p> <p>(?) Closetext; (?) End. (?) Writeln; (?) Eof (?) Close(F);</p>	ПК-02	32

125	<p>Каким оператором файл открывается для добавления ?</p> <p>(?) Delay (?) Append (?) Debug (?) Add (?) Rewrite</p>	ПК-02	32
126	<p>При работе с файлами обязательно надо описать ...</p> <p>(?) Константы (?) Файловую переменную (?) Строковую переменную (?) Подпрограмму (?) Функцию</p>	ПК-02	32
127	<p>Какой тип должен быть у файловой переменной (в разделе описания) ?</p> <p>(?) DOUBLE (?) STRING (?) TEXT (?) BOOLEAN (?) INTEGER</p>	ПК-02	32
128	<p>Какая строка соответствует описанию файла ?</p> <p>(?) Add (F, 'C:\pascal\file.pas') ; (?) File Open (F, 'C:\pascal\file.pas') ; (?) Append (F, 'C:\pascal\file.pas') ; (?) File (F, 'C:\pascal\file.pas') ; (?) Assign (F, 'C:\pascal\file.pas') ;</p>	ПК-02	33
129	<p>Для чего используется работа с файлами ?</p> <p>(?) Для облегчения работы ЭВМ (?) Так принято в Pascal (е) (?) Для программирования игр (?) Для освобождения оперативной памяти (?) Для записи результатов в файл, и чтения данных из файлов</p>	ПК-02	32
130	<p>При описании соответствия файловой переменной реальному файлу должен быть оператор</p> <p>(?) File Open (?) Append (?) Add (?) Assign (?) Watch</p>	ПК-02	32
131	<p>Каким оператором файл открывается для записи ?</p> <p>(?) Rewrite (?) Open (?) Output (?) Write (?) WriteLn</p>	ПК-02	32
132	<p>Каким оператором файл открывается для чтения ?</p> <p>(?) ReadLn (?) Read (?) Reset (?) ReadKey (?) Record</p>	ПК-02	32
133	<p>Какая строка должна быть при закрытии файла ?</p> <p>(?) Closetext; (?) End. (?) Writeln; (?) Eof (?) Close(F);</p>	ПК-02	32

134	<p>Чем отличается запись в файл от добавления ?</p> <p>(?) При записи файл создается, а при добавлении нет</p> <p>(?) Ничем</p> <p>(?) При записи идет обращение к подпрограмме, а при добавлении нет</p> <p>(?) При записи файл создается или стирает предыдущие записи, а при добавлении только создается или добавляет</p> <p>(?) Запись идет с начала строки, а добавление с конца первой строки</p>	ПК-02	32
135	<p>Современная система программирования, в основе которой лежит объектно-ориентированный язык Pascal и визуальные средства программирования, называется</p> <p>(?) Pascal</p> <p>(?) Basic</p> <p>(?) Delphi</p> <p>(?) Assembler</p>	ПК-02	32
136	<p>Что может называться компонентами Delphi?</p> <p>(?) Форма</p> <p>(?) Кнопки</p> <p>(?) Свойства объектов</p> <p>(?) Надписи</p>	ПК-02	32
137	<p>Форма, из которой вызываются все остальные зависимые формы, называется</p> <p>(?) заглавной</p> <p>(?) главной</p> <p>(?) главенствующей</p> <p>(?) основная</p>	ПК-02	32
138	<p>Проект -это</p> <p>(?) редактирование текста</p> <p>(?) настройка свойств среды программирования</p> <p>(?) процесс изменения программного кода, при котором выполняемые команды не меняются, но облегчается читаемость</p> <p>(?) совокупность файлов, из которых в результате должна получиться программа</p>	ПК-02	32
139	<p>Какая команда Главного меню, выполняет запуск проекта на выполнение:</p> <p>(?) Пуск</p> <p>(?) File</p> <p>(?) Run</p> <p>(?) Start</p>	ПК-02	32
140	<p>Какая команда Главного меню, позволяет выполнения различных операций над проектом:</p> <p>(?) Component</p> <p>(?) Run</p> <p>(?) Edit</p> <p>(?) Project</p>	ПК-02	32
141	<p>Вкладки окна Object Inspector:</p> <p>(?) Properties</p> <p>(?) Events</p> <p>(?) Standard</p> <p>(?) Win32</p>	ПК-02	32
142	<p>Какая функция обработки строк, преобразует целое число в строку:</p> <p>(?) StrToFloat(S)</p> <p>(?) Chr(x)</p> <p>(?) FloatToStr(S)</p> <p>(?) IntToStr (S)</p>	ПК-02	32
143	<p>Какая функция обработки строк, преобразует строку в вещественное число:</p> <p>(?) StrToFloat(S)</p> <p>(?) Chr(x)</p> <p>(?) FloatToStr(S)</p> <p>(?) IntToStr (S)</p>	ПК-02	32
144	<p>Файл для открытия проекта имеет расширение</p> <p>(?) .res</p> <p>(?) .dpr</p> <p>(?) .dfm</p> <p>(?) .pas</p>	ПК-02	32

145	Модуль формы проекта имеет расширение (?) .res (?) .dpr (?) .dfm (?) .pas	ПК-02	32
146	Программный модуль проекта имеет расширение (?) .res (?) .dpr (?) .dfm (?) .pas	ПК-02	32
147	Раздел описаний программного модуля (?) unit (?) interface (?) implementation (?) end	ПК-02	32
148	Раздел реализаций программного модуля (?) unit (?) interface (?) implementation (?) end	ПК-02	32
149	Для сохранения всех файлов проекта необходимо выбрать (?) save as (?) save all (?) save	ПК-02	32
150	Процесс выполнения программ на некотором наборе данных, для которого заранее известен результат применения или известны правила поведения этих программ, называется (?) Тестированием (?) Программированием (?) Интерпретацией (?) Разработкой алгоритма	ПК-02	32
151	Программным модулем в программировании называется (?) любой фрагмент описания процесса, оформленный как самостоятельный программный продукт пригодный для использования (?) любой фрагмент программного кода (?) любой самостоятельный программный продукт (?) программа или функционально завершенный фрагмент программы, предназначенный для: хранения, трансляции, объединения с другими частями программы и загрузки в оперативную память	ПК-02	32
152	Установите правильный порядок разработки программного модуля (?) изучение и проверка спецификации модуля, выбор языка, выбор алгоритма и структуры данных, программирование, шлифовка, проверка, компиляция модуля (?) изучение и проверка спецификации модуля, выбор языка, выбор алгоритма и структуры данных, программирование, проверка, компиляция модуля, шлифовка (?) выбор языка, выбор алгоритма и структуры данных, изучение и проверка спецификации модуля, программирование, шлифовка, проверка, компиляция модуля	ПК-02	32
153	Независимость от предыстории обращений к модулю, называется. (?) Рутинность (?) Структурированность (?) Параметром сцепления	ПК-02	32
154	Выберите из предложенного списка наилучший вид сцепления: (?) Сцепление по содержимому (?) Сцепление по общей области (?) Параметрическое сцепление	ПК-02	32
155	Какой вид сцепления модулей, рекомендуется для использования современной технологией программирования (?) Сцепление по содержимому (?) Сцепление по общей области (?) Параметрическое сцепление	ПК-02	32

156	<p>Выберите из предложенного списка понятия, относящиеся к критериям качества программного средства</p> <p>(?) функциональность</p> <p>(?) надежность</p> <p>(?) мобильность</p> <p>(?) достоверность</p> <p>(?) полезность (ценность)</p>	ПК-02	32
157	<p>Весь период разработки и эксплуатации (использования) , начиная от момента возникновения замысла ПС и кончая прекращением всех видов его использования, называется:</p> <p>(?) Программированием</p> <p>(?) Жизненным циклом</p> <p>(?) Отладкой</p>	ПК-02	32
158	<p>Процесс выполнения программ на некотором наборе данных, для которого заранее известен результат применения или известны правила поведения этих программ, называется</p> <p>(?) Тестированием</p> <p>(?) Программированием</p> <p>(?) Интерпретацией</p> <p>(?) Разработкой алгоритма</p>	ПК-02	32
159	<p>Программным модулем в программировании называется</p> <p>(?) любой фрагмент описания процесса, оформленный как самостоятельный программный продукт пригодный для использования</p> <p>(?) любой фрагмент программного кода</p> <p>(?) любой самостоятельный программный продукт</p> <p>(?) программа или функционально завершенный фрагмент программы, предназначенный для: хранения, трансляции, объединения с другими частями программы и загрузки в оперативную память</p>	ПК-02	32
160	<p>Установите правильный порядок разработки программного модуля</p> <p>(?) изучение и проверка спецификации модуля, выбор языка, выбор алгоритма и структуры данных, программирование, шлифовка, проверка, компиляция модуля</p> <p>(?) изучение и проверка спецификации модуля, выбор языка, выбор алгоритма и структуры данных, программирование , проверка, компиляция модуля, шлифовка</p> <p>(?) выбор языка, выбор алгоритма и структуры данных, изучение и проверка спецификации модуля, программирование, шлифовка, проверка, компиляция модуля</p>	ПК-02	32
161	<p>Процесс выполнения программ на некотором наборе данных, для которого заранее известен результат применения или известны правила поведения этих программ, называется</p> <p>(?) Тестированием</p> <p>(?) Программированием</p> <p>(?) Интерпретацией</p> <p>(?) Разработкой алгоритма</p>	ПК-02	32
162	<p>Модулем в программировании называется</p> <p>(?) любой фрагмент описания процесса, оформленный как самостоятельный программный продукт пригодный для использования</p> <p>(?) любой фрагмент программного кода</p> <p>(?) любой самостоятельный программный продукт</p> <p>(?) программа или функционально завершенный фрагмент программы, предназначенный для: хранения, трансляции, объединения с другими частями программы и загрузки в оперативную память</p>	ПК-02	32
163	<p>Порядок разработки программного модуля</p> <p>(?) изучение и проверка спецификации модуля, выбор языка, выбор алгоритма и структуры данных, программирование, шлифовка, проверка, компиляция модуля</p> <p>(?) изучение и проверка спецификации модуля, выбор языка, выбор алгоритма и структуры данных, программирование , проверка, компиляция модуля, шлифовка</p> <p>(?) выбор языка, выбор алгоритма и структуры данных, изучение и проверка спецификации модуля, программирование, шлифовка, проверка, компиляция модуля</p>	ПК-02	32

164	Компонент предназначенный для размещения на форме текстовой информации: (?) Edit (?) Label (?) Memo (?) ListBox	ПК-02	32
165	Компонет представляющий собой однострочное редактируемое текстовое поле: (?) CheckBox (?) RadioButton (?) Edit (?) Button	ПК-02	32
166	Компонент состоящий из группы взаимосвязанных индикаторов, из которых можно выбрать только один: (?) ComboBox (?) Memo (?) Label (?) RadioGroup	ПК-02	32
167	Компонент представляющий командную кнопку: (?) CheckBox (?) RadioButton (?) Edit (?) Button	ПК-02	32
168	Компонент представляет собой переключатель, который может быть находиться в одном из двух состояний: (?) CheckBox (?) RadioButton (?) Edit (?) Button	ПК-02	32
169	Компонент представляющий собой комбинацию списка и однострочного текстового поля: (?) ComboBox (?) StringGrid (?) Form (?) ListBox	ПК-02	32
170	Многостраничные панели позволяет создавать компонент: (?) ComboBox (?) StringGrid (?) Form (?) TPageControl	ПК-02	32
171	Компонент необходимый для визуального оформления формы: (?) CheckBox (?) RadioButton (?) GroupBox (?) Button	ПК-02	32
172	Признак доступности компонента (?) Checked (?) Enabled (?) Visible (?) Caption	ПК-02	32
173	Позволяет скрыть компонент (False) или сделать видимым (True) (?) Checked (?) Enabled (?) Visible (?) Caption	ПК-02	32
174	Доступ к базе данных в Delphi осуществляют компоненты: (?) ADOConnection (?) ADODataset (?) ADOTable (?) MainMenu	ПК-02	32

175	<p>В каком свойстве компонента ADOConnection содержится информация, необходимая для подключения к базе данных?</p> <p>(?) LoginPrompt (?) Connected (?) Mode (?) ConnectionString</p>	ПК-02	32
176	<p>В каком свойстве компонента ADODataSet прописывается SQL- команда?</p> <p>(?) Connection (?) CommandText (?) Activate (?) DataSet</p>	ПК-02	32
177	<p>На какой вкладке в Delphi находятся компоненты, обеспечивающие просмотр и редактирование полей, связанных с базами данных?</p> <p>(?) Data Controls (?) Win32 (?) DataAccess (?) Dialogs</p>	ПК-02	32
178	<p>Отображение данных в режиме таблицы обеспечивает компонент</p> <p>(?) TLabel (?) DBEdit (?) TEdit (?) TDBGrid</p>	ПК-02	32
179	<p>Отображение данных в режиме формы обеспечивает компонент</p> <p>(?) DBEdit (?) TEdit (?) TDBGrid (?) TLabel</p>	ПК-02	32
180	<p>Сколько записей отображается в режиме формы в окне программы</p> <p>(?) все записи таблицы (?) много (?) только одна</p>	ПК-02	32
181	<p>Какой минимальный набор команд должен находиться в SQL- запросе, обеспечивающий выборку данных из базы данных (таблицы)</p> <p>(?) SELECT СписокПолей FROM Таблица WHERE (Критерий) (?) SELECT СписокПолей FROM Таблица WHERE (Критерий) ORDER BY СписокПолей (?) DISTINCT список выбора FROM имя таблицы GROUP BY имя столбца (?) SELECT СписокПолей FROM Таблица</p>	ПК-02	32
182	<p>Какой оператор при использовании шаблонов в запросах среды Delphi вместо оператора = необходимо использовать</p> <p>(?) == (?) := (?) Like (?) WHERE</p>	ПК-02	32
183	<p>Использование какого оператора в SQL- запросах позволяет получить набор данных из базы данных без повторяющихся значений?</p> <p>(?) Distinct (?) From (?) Order By (?) Group By</p>	ПК-02	32
184	<p>После какого оператора указывается список полей, по которым производится группировка?</p> <p>(?) Distinct (?) From (?) Order By (?) Group By</p>	ПК-02	32

185	<p>Какой оператор используется для вывода данных на консоль. Выберите подходящие ответы из списка:</p> <p>(?) Console.ReadLine() (?) Console.Write() (?) Console.Read() (?) Console.WriteLine()</p>	ПК-02	32
186	<p>Какой оператор используется для ожидания нажатия клавиши в консоли.</p> <p>(?) Console.ReadLine() (?) Console.WriteRead() (?) Console.ReadKey() (?) Console.WriteLine()</p>	ПК-02	32
187	<p>Вывести на одной строке числа 5, 11, 1025 через пробел. Выберите все подходящие ответы из списка</p> <p>(?) Console.Write(5 + " 11 1025"); (?) Console.Write("5 11 " + 1025); (?) Console.WriteLine(5 + " " + 11 + 1025); (?) Console.WriteLine("5 11 1025"); (?) Console.Write("5 11 1025"); (?) Console.Write(5 + "11 1024");</p>	ПК-02	32
188	<p>Сопоставьте типы данных с их диапазоном значений:</p> <p>1. Целое число <u>а</u> char 2. Строка <u>б</u> bool 3. Логическое значение <u>в</u> double 4. Символ <u>г</u> int 5. Вещественное число <u>д</u> string</p>	ПК-02	32
189	<p>Что такое переменная? Выберите один вариант из списка:</p> <p>(?) Переменная — это ячейка памяти определенного типа, имеющая имя. (?) Переменная — это ячейка памяти. (?) Переменная — это ячейка памяти неопределенного типа, имеющая имя.</p>	ПК-02	32
190	<p>Чтобы записать в переменную значение, нужно воспользоваться следующим правилом. Выберите один вариант из списка:</p> <p>(?) Куда == Что (?) Куда = Что (?) Что = Куда (?) Что == Куда (?) Что := Куда</p>	ПК-02	32
191	<p>Как объявить и записать целое число 5 в переменную?</p> <p>(?) int n ==5; (?) int n = 5; (?) double n = 5; (?) bool n <= 5;</p>	ПК-02	32
192	<p>Какой оператор используется для ввода строки с консоли?</p> <p>(?) Console.Write() (?) Console.ReadLine() (?) Console.WriteLine() (?) Console.ReadKey()</p>	ПК-02	32
193	<p>Все данные, которые считываются с консоли, какой имеют тип данных?</p> <p>(?) string (?) bool (?) char (?) int (?) double</p>	ПК-02	32

194	<p>Как перевести переменную строкового типа "myString" в целый числовой тип данных?</p> <pre>string myString="10"; (?) int number = Convert.ToDouble(myString); (?) int number = Convert.ToBoolean(myString); (?) int number = Convert.ToInt32(myString); (?) int number = Convert.ToString(myString);</pre>	ПК-02	32
195	<p>Определите значение переменной result после выполнения фрагмента программы:</p> <pre>int num1 = 15; num1 = num1 + 6; int num2 = -num1; int result = -num1 - 2 * num2;</pre> <p>(?) 63 (?) -63 (?) 21 (?) -21</p>	ПК-02	32
196	<p>Определите значение переменной result после выполнения с фрагмента программы:</p> <pre>int num1 = 10 + 3 * 5; int num2 = (num1 % 10) + 14; num1 = (num2 / 10) + 3; int result = num1 - num2;</pre>	ПК-02	32
197	<p>Определите значение переменной result после выполнения фрагмента программы:</p> <pre>int num1 = 20; num1 = num1 + 23; int num2 = num1 / 4 - num1 / 2; int result = num1 - num2;</pre>	ПК-02	32
198	<p>Определите значение переменной operand после выполнения фрагмента программы:</p> <pre>int operand = 1905; int y = operand / 100; operand = (operand % 100) * 10; operand = operand + y;</pre> <p>(?) 19069,5 (?) 209 (?) 69 (?) 209,5</p>	ПК-02	32
199	<p>Выберите оператор, который возвращает число a, округленное до определенного количества знаков после запятой, представленного параметром b</p> <p>(?) Pow(double a, double b) (?) Round(double a, round b) (?) Round(double b) (?) BigMul(int a, int b)</p>	ПК-02	32

200	<p>Проанализируйте следующий код программы и определите значение переменных a и s :</p> <pre> int s = 1; int a = 5; if (s > 1) { a = a + s; } else { s = s + a * 2; } (?) a=6; s=1; (?) a=5; s=11; (?) a=5; s=1; (?) a=6; s=11; </pre>	ПК-02	32
201	<p>Определите значение переменной "c" после выполнения фрагмента программы:</p> <pre> int a = 30; int b = 6; a = a / 2 * b; int c; if (a > b) { c = a - 3 * b; } else { c = a + 3 * b; } </pre>	ПК-02	32
202	<p>Определите значение переменной "c" после выполнения фрагмента программы. Ответ запишите с точностью до целого.</p> <pre> int a = 56; int b = 7; int k = 6; a = a / 7 - b; int c; if (a > b) { c = a - k * b; } else { c = a + k * b; } </pre>	ПК-02	32
203	<p>Найдите значение переменных resultA, resultB, resultC:</p> <pre> int a = 10; int b = 20; bool c = true; bool d = false; bool resultA = (a > 5) && (b > 5) && (a < 20) && (b < 30); bool resultB = !(a < 15) !(b < 30); bool resultC = c d && (b == 20);□ (?) resultA =true; resultB =false; result= false; (?) resultA =true; resultB = true; result=true; (?) resultA =true; resultB =false; result=true; (?) resultA = false; resultB =false; result=true; </pre>	ПК-02	32

204	<p>В каком порядке работает цикл for?</p> <p>1 инициализация счетчика 2 условие входа 3 изменения переменной цикла</p>	ПК-02	32
205	<p>Сколько итераций цикла в данной программе?</p> <pre>for (int i = -25; i <= 25; i++) { }</pre> <p>(?) 51 (?) 0 (?) 1 (?) 50</p>	ПК-02	32
206	<p>Определите, сколько раз и как сработает следующая программа?</p> <pre>for (int i = 5; i < 20; i++) { Console.WriteLine("Hello World!"); }</pre> <p>(?) 15 раз выведется фраза "Hello World!" в одной строке (?) 15 раз выведется фраза "Hello World!", каждый раз на новой строке (?) 14 раз выведется фраза "Hello World!" в одной строке (?) 14 раз выведется фраза "Hello World!", каждый раз на новой строке</p>	ПК-02	32
207	<p>Сопоставьте циклы, которые выполняют одинаковое количество итераций:</p> <p>1. for (int i = -2; i < 9; i++) a for (int j = -1; j <= 9; j++) 2. for (int b = 0; b < 17; b++) b for (int k = 5; k >= -11; k--) 3. for (int y = 18; y > -1; y--) c for (int i = 19; i <= 37; i++) 4. for (int h = -10; h < -2; h++) d for (int x = 8; x > 0; x--)</p>	ПК-02	32
208	<p>Определите, какую задачу решает следующий код?</p> <pre>int sum = 0; for (int i = 19; i > 1; i--) sum += i; Console.WriteLine(sum);</pre> <p>(?) программа находит сумму чисел от 2 до 19 (?) программа находит сумму чисел от 19 до 2 (?) программа находит сумму чисел от 1 до 19 (?) программа находит сумму чисел от 19 до 1</p>	ПК-02	32
209	<p>Проанализируйте предложенный код и выберите, какую задачу он решает.</p> <pre>int count = 0, int.Parse(Console.ReadLine()); for (int i = 1; i <= n; i++) if (i % 2 == 0 && i % 5 != 0) count++; Console.WriteLine(count);</pre> <p>(?) Находит количество четных чисел от 1 до n не кратных 5. (?) Находит количество нечетных чисел от 1 до n не кратных 5. (?) Находит количество четных чисел от 1 до n кратных 5. (?) Находит количество нечетных чисел от 1 до n не кратных 5.</p>	ПК-02	32
210	<p>Определите, что выведет код?</p> <pre>int number = 10; while (number < 23) { Console.Write(number + "; "); number += 3; }</pre> <p>(?) 10; 13; 16; 19; 21; 23; (?) 13; 16; 19; 21; 23; (?) 13; 16; 19; 21; (?) 10; 13; 16; 19; 22;</p>	ПК-02	32

211	<p>Определите, что выведет код?</p> <pre>int number = 257; int digit; while (number > 0) { digit = number % 10; number /= 10; Console.Write(digit + "; "); } (?) 7; 5; 2; (?) 752 (?) 257 (?) 2;5;7;</pre>	ПК-02	32
212	<p>Определите, что выведет код?</p> <pre>int number = 27; int i = 9; int cnt = 0; while (i <= number) { if (i % 2 == 0) cnt ++; i++; } Console.WriteLine(cnt); (?) 9 (?) 10 (?) 8 (?) 11</pre>	ПК-02	32
213	<p>Определите, что выведет код:</p> <pre>int p = 1, i = 2, n = int.Parse(Console.ReadLine()); while (i <= n) { p = p * i; i++; } Console.WriteLine(p); (?) произведение всех чисел от 2 до n (?) произведение всех чисел от 2 до n с шагом 2 (?) произведение всех чисел от n до 2 (?) произведение всех чисел от n до 2 с шагом 2</pre>	ПК-02	32
214	<p>Определите, что выведет фрагмент программы?</p> <pre>int number = 6878; int cnt = 1; while (number > 0) { cnt ++; number /= 10; } Console.WriteLine(cnt); (?) 4 (?) 10 (?) 5 (?) 8786</pre>	ПК-02	32

215	<p>Определите, что выведет фрагмент программы?</p> <pre>int number = 3498; int cnt = 1; while (number > 0) { Console.Write(number % 10 + " "); number /= 10; }</pre> <p>(?) 8943 (?) 8 9 4 3 (?) 4 (?) 40</p>	ПК-02	32
216	<p>Определите, какое значение будет иметь переменная summa после выполнения фрагмента программы:</p> <pre>int summa = 6; for (int i = 2; i < 22; i++) { if (i % 5 == 0) break; summa += i; }</pre> <p>(?) 10 (?) 4 (?) 28 (?) 15</p>	ПК-02	32
217	<p>Определите, какое значение будет иметь переменная summa после выполнения фрагмента программы:</p> <pre>int summa=0; int divider=10; for (int i = 10; i > 1; i--) { if (divider % i == 0) continue; summa += i; }</pre> <p>(?) 37 (?) 42 (?) 40 (?) 35</p>	ПК-02	32
218	<p>Определите, какое значение будет иметь переменная i после выполнения фрагмента программы:</p> <pre>int i = 0, summa = 0; while (i + 1 < 10) { i += 2; summa += i; if (summa > 16) break; }</pre> <p>(?) 10 (?) 6 (?) 8 (?) 24</p>	ПК-02	32

219	<p>Определите, что выведет фрагмент программы? string user = "Serg Sorokin"; Console.WriteLine(user.Length);</p> <p>(?) 12 (?) 11 (?) Serg Sorokin (?) Ошибка</p>	ПК-02	32
220	<p>Определите, что выведет фрагмент программы? string user = "Serg Sorokin"; Console.WriteLine(user.Substring(5,7));</p> <p>(?) Sorokin (?) rokin (?) «пробел» Sorokin (?) oroki</p>	ПК-02	32
221	<p>Определите, что выведет следующий фрагмент программы? string user = "Serg Sorokin" - "Sorokin";</p> <p>(?) Serg (?) произойдет вызов не компилируемой ошибки (?) Serg «пробел» (?) Sorokin</p>	ПК-02	32
222	<p>Дана строковая переменная: string hashTag = "#WriteCode";</p> <p>Определите соотношение индексов к символам, для обозначения ошибки использовать Егг. Ответы перечисляйте через пробел без запятых и точек -21947 <input type="checkbox"/></p>	ПК-02	32
223	<p>Дана строковая переменная: string hashTag = "#WriteCode";</p> <p>Определите соотношение индексов к символам, для обозначения ошибки использовать Егг. Ответы перечисляйте через пробел без запятых и точек 528120 <input type="checkbox"/></p>	ПК-02	32
224	<p>С какого индекса начинается нумерация массива? (?) 1 (?) 0 (?) 100 (?) -1</p>	ПК-02	32
225	<p>Сопоставьте элемент массива с его индексом: short[] numbers = new short[] {23, 78, 121, 87, 975, 100, 0, 54, 19, 0}; Ответы перечисляйте через пробел без запятых и точек [5][2][7][1][9] <input type="checkbox"/></p>	ПК-02	32
226	<p>Сопоставьте элемент массива с его индексом: short[] numbers = new short[] {23, 78, 121, 87, 975, 100, 0, 54, 19, 0}; Ответы перечисляйте через пробел без запятых и точек [4][6][0][3][8] <input type="checkbox"/></p>	ПК-02	32
227	<p>Дан массив чисел из n элементов. Какой индекс будет у последнего элемента массива? (?) n+1 (?) 0 (?) n-1 (?) n</p>	ПК-02	32

228	<p>Рассмотрите следующий фрагмент кода. Определите, что выведет в консоль приведенный ниже фрагмент:</p> <pre>int[] array = {19, 15, 15, 13, 14, 5, 7, 9, 4, 19}; int cnt = 0; for (int i = 0; i < 9; i++) if (array[i + 1] >= array[i]) { int tmp = array[i]; array[i] = array[i - 1]; array[i - 1] = tmp; } else cnt ++; Console.WriteLine("Количество = " + cnt);</pre> <p>(?) 4 (?) 5 (?) сообщение об ошибке (?) 19,4,9,7,5,14,13,15,15,19</p>	ПК-02	32
229	<p>. Определите, что выведет в консоль приведенный ниже фрагмент:</p> <pre>int[] array = {19, 15, 15, 13, 14, 5, 7, 9, 4, 19}; int sum = 0; for (int i = 2; i < array.Length; i++) sum += array[i] - array[i - 2]; Console.WriteLine("sum = " + sum);</pre>	ПК-02	32
230	<p>Определите, что выведет в консоль приведенный ниже фрагмент:</p> <pre>int[] array = {18, 10, 10, 7, 8, -9, -5, -2, -12, 18}; int i = 0, j = 0; while (array[i] <= 0) i++; while (array[j] >= 0) j++; int result = array[i] - array[j]; Console.WriteLine("result = " + result);</pre>	ПК-02	32
231	<p>Определите, что выведет в консоль приведенный ниже фрагмент:</p> <pre>int[] array = {28, 11, 28, 46, 16, 16, 43, 48, 23, 23}; int sum = 0; for (int i = 1; i < 10; i++) if (array[i - 1] < array[i]) { array[i] = array[i - 1] + 1; sum++; } Console.WriteLine("sum = " + sum);</pre>	ПК-02	32
232	<p>Укажите правильное объявление двумерного массива.</p> <p>(?) int[,] array = new int[,]; (?) double[14, 17] debetDay = new double[,]; (?) string[,] fio = new string[-3, 100]; (?) int[,] таблица = new int[1, 10]; (?) double[,] table = new double[2, 7]; (?) string[,] for = new string[3, 100]; (?) int[,] array = new double[10, 10];</p>	ПК-02	32
233	<p>Положение элемента двумерного массива определяется:</p> <p>(?) порядковым номером элемента в таблице; (?) размерностью элемента в таблице; (?) номером строки элемента в таблице; (?) номерами строки и столбца элемента в таблице; (?) номером столбца элемента в таблице.</p>	ПК-02	32

234	<p>Выберите условие принадлежности элемента двумерной таблицы главной диагонали:</p> <p>(?) $i \geq j$ (?) $i > j$ (?) $i != j$ (?) $i = j$ (?) $i < j$ (?) $i == j$ (?) $i \leq j$</p>	ПК-02	32
235	<p>Сопоставьте условиям соответствующие им признаки.</p> <p>Примечание: i - номер строки, j - номер столбца.</p> <p>а). $i > j$ 1). выше главной диагонали б). $i < j$ 2). ниже главной диагонали в). $i == j$ 3). главная диагональ</p>	ПК-02	32
236	<p>Определите, что находит описанный ниже алгоритм:</p> <pre>int summa = 0; for (int i = 0; i < array.GetLength(0); i++) for (int j = 0; j < array.GetLength(1); j++) if (array[i, j] >= 0) summa += array[i, j];</pre> <p>(?) минимальный элемент массива (?) сумму неотрицательных элементов массива (?) количество отрицательных элементов массива (?) максимальный элемент массива (?) сумму неотрицательных элементов массива (?) индекс последнего отрицательного элемента массива (?) ни один из ответов не верен</p>	ПК-02	32
237	<p>Укажите тип возвращаемого значения следующего метода:</p> <pre>static char ReturnSymbol(string value, int pos, double divider)</pre> <p>(?) int (?) double (?) char (?) void (?) string</p>	ПК-02	32
238	<p>Укажите какие методы объявлены корректно:</p> <p>(?) static int void ToInt(string value) (?) static int Summa(int a, b) (?) static int long MaxSummDigits(int number) (?) static bool IsHaveSymbol(string text, char symbol) (?) static int Difference(a, b)</p>	ПК-02	32
239	<p>Укажите какие методы объявлены не корректно:</p> <p>(?) static void ToInt(string value[]) (?) static void Summa(int[] a, b) (?) static int[] MaxSummDigits(number int) (?) static boolean IsHaveSymbol(char symbol) (?) static void Difference(int b, int c)</p>	ПК-02	32

240	<p>Укажите какие вызовы метода корректны: static double Diffusion(double a, double b) { □ return ...; } (?) string text = Diffusion(0.15, 0.25); (?) double diffusion = Diffusion(24.85, 30.1); (?) double diffusion = Diffusion(18.8, 17.1, 19.5); (?) Console.WriteLine("Diffusion equals: " + Diffusion(14.4, "number")); (?) Console.WriteLine("Diffusion = " + Diffusion(0.4)); (?) Console.WriteLine("diffusion == " + Diffusion(14.4, 45)); (?) Console.WriteLine("this difusion = " + Difusion(26.2, 0));</p>	ПК-02	32
241	<p>Какие языки доступны для программирования на платформе .NET при установке среды Visual Studio? (?) PHP (?) C# (?) Visual Basic (?) Pascal</p>	ПК-02	32
242	<p>На каких ОС можно создавать приложения с помощью платформы .NET? (?) Windows (?) Linux (?) Android (?) MacOS</p>	ПК-02	32
243	<p>Какие сервисы предоставляет платформа .NET Framework для управления памятью? (?) CLR (Common Language Runtime) (?) CTS (Common Type System) (?) Программист сам управляет памятью при разработке ПО</p>	ПК-02	32
244	<p>Как можно записать увеличение значения переменной на 1 с помощью инкрементного оператора? (?) var i = 0; i++; (?) var i = 0; i = i+1; (?) var i = 0; ++i; (?) var i = 0; i=++i;</p>	ПК-02	32
245	<p>Какие значения могут принимать данные логического типа (Bool)? (?) false (?) true (?) "False" (?) "True"</p>	ПК-02	32
246	<p>Какие операции отношения можно использовать для сравнения значений? (?) <, <=, >, >=, ==, != (?) <, <=, >, >=, =, <> (?) !!, ^^, &&, (?) Ничего из перечисленного</p>	ПК-02	32
247	<p>Какие типы данных используются для целых чисел в C#? (?) int (?) long (?) real (?) short (?) byte</p>	ПК-02	32
248	<p>Как объявить переменную с десятичным типом данных в C#? (?) float x; (?) decimal x; (?) double x; (?) ushort x;</p>	ПК-02	32

249	<p>В чем разница между типами данных float и double?</p> <p>(?) float и double оба вещественные типы данных, но для переменной float выделяется 8 байтов памяти, а для double 16</p> <p>(?) float и double оба вещественные типы данных, но для переменной float выделяется 4 байтов памяти, а для double 8</p> <p>(?) float и вещественный тип данных, в то время как double – десятичный тип данных</p> <p>(?) double и вещественный тип данных, в то время как float – десятичный тип данных</p> <p>Какие операции доступны для работы с вещественными числами в C#?</p> <p>(?) умножение</p> <p>(?) сложение</p> <p>(?) урезание</p>	ПК-02	32
250	<p>Выберите правильное утверждение</p> <p>(?) Цикл с предусловие проверяет условие перед выполнением тела итерации</p> <p>(?) Цикл с постусловием сначала выполняет итерацию, а потом проверяет условие выхода из цикла</p> <p>(?) В C# есть только цикл с предусловием while (true false) {}</p> <p>(?) Если неправильно выбрано условие, то цикл может образоваться бесконечный цикл</p>	ПК-02	32
251	<p>Что произойдет, если выражение останется ложным при первой проверке в цикле while?</p> <p>(?) Цикл перейдет к следующей итерации</p> <p>(?) Цикл войдет в бесконечное выполнение</p> <p>(?) Цикл завершит свое выполнение</p>	ПК-02	32
252	<p>Какова структура цикла с параметром for?</p> <p>(?) for (int i = 0; i < 10; i++)</p> <p>(?) for (int i = 0; i <= 10; j++)</p> <p>(?) for (var i = 0; i <= 10; i++)</p> <p>(?) for (int i, i < 10, i++)</p>	ПК-02	32
253	<p>Какой код выведет на экран надпись «I am a KING»</p> <p>(?) Console.WriteLine("I am a KING");</p> <p>(?) Console.WriteLine("I am a KING");</p> <p>(?) // Console.WriteLine("I am a KING");</p> <p>(?) console.writeline("I am a KING")</p> <p>(?) Console.Writeline('I am a KING');</p>	ПК-02	32
254	<p>Что произойдет в цикле do while, если условие всегда будет оставаться истинным?</p> <p>(?) Тело цикла будет выполняться бесконечно</p> <p>(?) Цикл выполнится один раз и завершит свое выполнение</p> <p>(?) Тело цикла не выполнится ни разу</p> <p>(?) Цикл будет выполняться до тех пор, пока внутри тела цикла не встретится оператор break</p>	ПК-02	32
255	<p>Почему в теле цикла должны находиться операторы, приводящие к изменению значения выражения?</p> <p>(?) В цикле происходит вычисление какого-то параметра и нужны переменные</p> <p>(?) Изменение выражения нужно, чтобы в какой-то момент времени условие выхода сработало и осуществился выход из цикла</p> <p>(?) Не обязательно использовать операторы, которые изменяют значение выражения</p>	ПК-02	32
256	<p>Что делает метод Array.FindIndex()?</p> <p>(?) Осуществляет поиск индекс элемента</p> <p>(?) Осуществляет поиск элемента по указанному индексу</p> <p>(?) Осуществляет вывод всех элементов массива до указанного индекса</p> <p>(?) Осуществляет вывод всех элементов массива, начиная с указанного индекса</p>	ПК-02	32
257	<p>Имеется следующий код: char symbolOne = 'b'; char symbolTwo = 'w'. Укажите все варианты, где условие будет TRUE</p> <p>(?) symbolOne == symbolTwo</p> <p>(?) symbolOne > symbolTwo</p> <p>(?) symbolOne < symbolTwo</p> <p>(?) symbolOne < 'q'</p> <p>(?) symbolOne > '%'</p>	ПК-02	32

258	Как получить числовой код символа в C#? (?) char symbol = 'ш'; int code = (int) symbol; (?) char symbol = 'ш'; int code = Convert.ToInt32(symbol); (?) char symbol = 'ш'; int code = symbol.ToInt(); (?) char symbol = 'ш'; int code = Convert.ToChar(symbol);	ПК-02	32
-----	---	-------	----

5.3. Материалы для оценки достижения компетенций

5.3.7. Вопросы для устного опроса

№	Содержание	Компетенция	ИД
1	Что делает оператор «%»	ПК-02	32
2	Как сделать инкрементацию числа .	ПК-02	32
3	Как сделать декрементация числа	ПК-02	32
4	Как найти квадратный корень из числа x	ПК-02	33
5	Обозначения оператора «НЕ»	ПК-02	32
6	Обозначение оператора «ИЛИ»	ПК-02	32
7	Обозначение оператора «И»	ПК-02	32
8	Как называется оператор «?:»	ПК-02	32
9	Что такое массив	ПК-02	32
10	Какие бывают массивы ?	ПК-02	32
11	Что такое цикл и для чего они нужны	ПК-02	32
12	Какие бывают циклы?	ПК-02	32
13	Какой оператор возвращает значение из метода ?	ПК-02	32
14	Что такое константа ?	ПК-02	32
15	Что обозначает ключевое слово var ?	ПК-02	32
16	Что такое Куча ?	ПК-02	32
17	Как правильно объявляется неявно типизированные переменные?	ПК-02	32
18	Для того чтобы вывести на экран какое-либо сообщение воспользуйтесь процедурой?	ПК-02	33
19	Отличительной особенностью каких языков программирования является их ориентация не на систему команд той или иной ЭВМ а на систему операторов характерных для записи определенного класса алгоритмов?	ПК-02	32
20	Что используют все языки программирования высокого уровня для предоставления программисту простого и легкого доступа к различным объектам?	ПК-02	32
21	Что понимают под языком программирования (ЯП)?	ПК-02	32
22	Как называется именованная спецификация одного или более столбцов (для каждого столбца указывается имя а также его тип или домен)?	ПК-02	32
23	Что понимается под наследованием типов?	ПК-02	32
24	Что представляет собой открытый массив?	ПК-02	32
25	Каким способом параметр массив всегда передается в функцию?	ПК-02	32
26	Что понимается под классом в объектно-ориентированном программировании (ООП)?	ПК-02	32
27	Какие синтаксические правила объявления переменных являются корректными:	ПК-02	32
28	Какие типы в языке C# относятся к значимым:	ПК-02	32
29	Какие типы в языке C# относятся к ссылочным:	ПК-02	32
30	Какой метод класса Console выводит символьную строку дополняя её в конце символами перехода на новую строку и возврата каретки?	ПК-02	33
31	Какой метод класса Console считывает с системной консоли один символ?	ПК-02	33
32	Какой метод класса Console позволяет считать с системной консоли информацию до ближайшего символа перехода на новую строку?	ПК-02	33
33	Результатом какой операции является значение операнда после его увеличения?	ПК-02	32
34	Укажите ссылочные типы	ПК-02	32
35	Укажите ВСЕ значимые типы	ПК-02	32
36	Целочисленными типами являются (перечислите ВСЕ целочисленные типы):	ПК-02	32
37	Когда задается начальное значение переменной цикла в программе	ПК-02	32
38	К каким операторам относится оператор break	ПК-02	32
39	К каким операторам относится оператор continue	ПК-02	32
40	С помощью какого оператора можно в любой момент прервать выполнение цикла?	ПК-02	32
41	Сколько раз и где выполняется инициализация переменной цикла в операторе for?	ПК-02	32
42	Оператор while проверяет условие завершения цикла	ПК-02	32
43	Как оператор while изменяет значения переменной цикла?	ПК-02	32
44	Можно ли в цикле while использовать прерывания цикла break и возобновления цикла continue	ПК-02	32
45	Когда проверяется условие завершения цикла в Операторе do?	ПК-02	32
46	Можно ли в цикле do использовать прерывания цикла break и возобновления цикла continue	ПК-02	32
47	Для обработки каких типов данных используется Оператор foreach?	ПК-02	32
48	Нужна ли переменная цикла и нужно ли её инициализировать инкрементировать и проверять не вышло ли значение индекса массива за допустимые пределы в операторе foreach?	ПК-02	32
49	Что возвращает оператор is зависимости от того можно ли выполнить преобразование типов или нет?	ПК-02	33

50	Укажите лишний оператор в перечислении:	ПК-02	32
51	Какое количество экземпляров case может включать оператор switch?	ПК-02	32
52	Какого типа должно быть «Выражение» в операторе If (<Выражение>) <Оператор 1> [else <Оператор 2>] ?	ПК-02	33
53	Какой из предложенных шаблонов в Visual Studio предназначен для создания оконных приложений?	ПК-02	33
54	Как называется окно в котором отображаются файлы проекта	ПК-02	32
55	Какие из перечисленных кнопок не принадлежат к кнопкам управления окна Solution Explorer	ПК-02	32
56	Какие интерфейсные элементы позволяют редактировать текст	ПК-02	32
57	Какие интерфейсные элементы позволяют объединять другие элементы управления	ПК-02	32
58	Какие интерфейсные элементы позволяют вводить значения	ПК-02	32
59	Какие интерфейсные элементы предназначены для хранения графики	ПК-02	32
60	Какие интерфейсные элементы предназначены для взаимодействия с пользователем	ПК-02	32
61	Какие интерфейсные элементы предназначены для работы с датами	ПК-02	32
62	Какой элемент управления нужно использовать чтобы выбрать файл с диска во время работы программы:	ПК-02	32
63	Где можно просмотреть все свойства и события для выбранного интерфейсного элемента:	ПК-02	32
64	Где осуществляется написание программного кода	ПК-02	У2
65	Какие свойства элемента управления наследуются от содержащего его контейнера	ПК-02	32
66	Какие свойства отвечают за размещение визуального элемента на форме	ПК-02	32
67	Какие свойства определяют надпись или текст в элементе управления, а также направление текста и способ его редактирования	ПК-02	32
68	Какие свойства определяют доступен ли пользователю интерфейсный элемент и отображается ли он	ПК-02	32
69	Укажите основные события для элемента управления «Button»	ПК-02	32
70	Для чего предназначен элемент управления DataGridView	ПК-02	32
71	Сколько фрагментов текста или иных элементов проектов может хранить буфер обмена в Visual Studio	ПК-02	32
72	Как запустить отладку приложения в Visual Studio	ПК-02	32
73	В каком окне будут отображаться ошибки компиляции приложения	ПК-02	32
74	Для чего необходима обработка исключений	ПК-02	32
75	Какой блок в try..catch..finally является обязательным	ПК-02	32
76	Может ли обработка исключений содержать в себе несколько блоков try	ПК-02	32
77	Может ли обработка исключений содержать в себе несколько блоков Finally	ПК-02	32
78	Укажите базовый класс для всех исключений в C#	ПК-02	32
79	Назовите свойства базового класса Exception	ПК-02	32
80	Какой оператор позволяет сгенерировать исключение в процессе работы приложения	ПК-02	32
81	Для чего необходима отладка приложения	ПК-02	32
82	Какие горячие клавиши позволяют запустить приложение на отладку:	ПК-02	32
83	Какая горячая клавиша позволяет пошагово выполнять алгоритм в процессе отладки без захода внутрь методов	ПК-02	32
84	Какая горячая клавиша позволяет пошагово выполнять алгоритм в процессе отладки с заходом внутрь методов	ПК-02	32
85	Как можно проверить значение переменной в процессе выполнения приложения	ПК-02	32

5.3.8. Задания для проверки формирования навыков

№	Содержание	Компетенция	ИД
1	Создание консольного приложения, определяющего тип объект (значимый или ссылочный)	ПК-02	Н2
2	Создание консольного приложения, используя функцию сортировки строк	ПК-02	Н2
3	Создание консольного приложения, обрабатывающее событие, генерируемое таймером	ПК-02	Н2
4	Разработка консольного приложения, создающего файл и записывающего в него текст	ПК-02	Н2
5	Разработка консольного приложения, выводящее содержимое файла на экран	ПК-02	Н2
6	Создание консольного приложения, способного переводить текстовый файл из одной кодировки в другую	ПК-02	Н3
7	Создание консольного приложения, обрабатывающее файл и отображающее время, дату и коды завершения	ПК-02	Н2
8	Создание консольного приложения, принимающее имя, адрес и телефон пользователя проверяющее ввод данных с помощью регулярных выражений	ПК-02	Н2
9	Создание консольного приложения, читающее файл и изменяющее формат даты на "месяц-день-год" и записывающее результаты в новый файл	ПК-02	Н3
10	Создание консольного приложения для поиска заданного файла на диске	ПК-02	Н3
11	Создание консольного приложения, позволяющее просматривать файл в текстовом окне	ПК-02	Н3

5.3.9. Вопросы для контрольной (расчетно-графической) работы

Не предусмотрено

5.4. Система оценивания достижения компетенций

5.4.1. Оценка достижения компетенций в ходе промежуточной аттестации

Индикаторы дотижения компетенций		Номера вопросов и задач		
Код	Содержание	вопросы к экзамену	задачи к экзамену	вопросы к зачету
ПК-02 Способность разрабатывать и адаптировать прикладное программное обеспечение				
З2	лексические, синтаксические и семантические правила языка программирования	1-22, 26		1-16, 18
З3	функциональные возможности систем программирования	23-25, 27		
У2	использовать языки программирования для решения профессиональных задач			17, 19-22
У3	разрабатывать программы с использованием систем программирования		2-3, 8, 9	
Н2	разработки программ на языках программирования высокого уровня		4-7, 10-17	
Н3	использования систем программирования при решении профессиональных задач		1	

5.4. Система оценивания достижения компетенций
5.4.2. Оценка достижения компетенций в ходе текущего контроля

Индикаторы дотижения компетенций		Номера вопросов и задач		
Код	Содержание	вопросы тестов	вопросы устного опроса	задачи для проверки навыков
ПК-02 Способность разрабатывать и адаптировать прикладное программное обеспечение				
32	лексические, синтаксические и семантические правила языка программирования	1, 2, 4, 5, 7-9, 13-29, 31-35, 37-39, 41-59, 63, 64, 67-72, 74, 76-80, 82-90, 92, 93, 96, 97, 103-113, 115-258	1-3, 5-17, 19-29, 33-48, 50, 51, 54-63, 65-85	
33	функциональные возможности систем программирования	3, 6, 10, 11, 30, 36, 60-62, 91, 98, 102	4, 18, 30-32, 49, 52, 53	
У2	использовать языки программирования для решения профессиональных задач	40, 65, 81, 95, 114	64	
У3	разрабатывать программы с использованием систем программирования	12, 66, 73, 75, 94, 99-101		
Н2	разработки программ на языках программирования высокого уровня			1-5, 7, 8
Н3	использования систем программирования при решении профессиональных задач			6, 9-11

6. Учебно-методическое обеспечение дисциплины

6.1. Рекомендуемая литература

№	Библиографическое описание	Вид издания
1	Бедердинова О. И. Программирование на языках высокого уровня [электронный ресурс]: Учебное пособие / О. И. Бедердинова, Т. А. Минеева - Москва: ООО "Научно-издательский центр ИНФРА-М", 2019 - 159 с. [ЭИ] [ЭБС Знаниум] URL: http://znanium.com/catalog/document?id=344897	Учебное
2	Гуриков С. Р. Введение в программирование на языке Visual C# [электронный ресурс]: Учебное пособие / С. Р. Гуриков - Москва: Издательство "ФОРУМ", 2020 - 447 с. [ЭИ] [ЭБС Знаниум] URL: http://znanium.com/catalog/document?id=338986	Учебное
3	Копырин, А. С. Программирование на C# в Visual Studio 2013 : учебное пособие / А. С. Копырин, Т. Л. Салова. - Москва : ФЛИНТА, 2021. - 54 с. - ISBN 978-5-9765-4754-4. - Текст : электронный. - URL: https://znanium.com/catalog/product/1851994	Учебное
4	Хорев, П. Б. Объектно-ориентированное программирование с примерами на C# : учебное пособие / П.Б. Хорев. — Москва : ФОРУМ : ИНФРА-М, 2023. — 200 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-00091-680-3. - Текст : электронный. - URL: https://znanium.com/catalog/product/1926392	Учебное
5	Улезько А.В. Порядок оценивания результатов достижения компетенций: методические материалы для основной образовательной программы по направлению: 09.03.03 Прикладная информатика, профиль: Информационные системы и технологии в менеджменте АПК / А.В. Улезько, С.А. Кулев, А.А. Толстых. – Воронеж: ВГАУ, 2019. – 24 с.	Методическое
6	Улезько А. В. Порядок формирования компетенций: методические материалы для основной образовательной программы бакалавриата по направлению: 09.03.03 Прикладная информатика, профиль: Информационные системы и технологии в менеджменте АПК / А.В. Улезько, С.А. Кулев, А.А. Толстых. – Воронеж: ВГАУ, 2019. – 39 с	Методическое
7	Программирование: журнал / Учредители: Российская академия наук, Московский государственный университет - Москва: Российская академия наук, 2020 [ЭИ] URL: https://elibrary.ru/contents.asp?titleid=7966	Периодическое
8	Программные продукты и системы: международный научно-практический журнал / Издатель: Научно-исследовательский институт Центрпрограммсистем ; учредитель: В. П. Куприянов - Тверь: Научно-исследовательский институт Центрпрограммсистем, 2020 [ЭИ] URL: https://elibrary.ru/contents.asp?titleid=9834	Периодическое

6.2. Ресурсы сети Интернет

6.2.1. Электронные библиотечные системы

№	Название
1	Лань
2	ZNANIUM.COM
3	ЮРАЙТ
4	IPRbooks
5	E-library
6	Электронная библиотека ВГАУ

6.2.2. Профессиональные базы данных и информационные системы

№	Название	Размещение
1	Справочная правовая система Гарант	http://www.consultant.ru/
2	Справочная правовая система Консультант Плюс	http://ivo.garant.ru

6.2.3. Сайты и информационные порталы

№	Название	Размещение
1	Интернет-технологии.ру	https://www.internet-technologies.ru
2	Calltouch	https://www.calltouch.ru
3	Портал открытых данных РФ	https://data.gov.ru/

7. Материально-техническое и программное обеспечение дисциплины

7.1. Помещения для ведения образовательного процесса и оборудование

№	Наименование помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом, в том числе помещения для самостоятельной работы, с указанием перечня основного оборудования, учебно-наглядных пособий и используемого программного обеспечения	Адрес (местоположение) помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом (в случае реализации образовательной программы в сетевой форме дополнительно указывается наименование организации, с которой заключен договор)
1	Учебная аудитория для проведения занятий: комплект учебной мебели, демонстрационное оборудование, учебно-наглядные пособия в виде презентаций, программное обеспечение: MS Windows /Linux /Ред ОС	394087, Воронежская область, г. Воронеж, ул. Мичурина, д.1
2	Учебная аудитория для проведения учебных занятий: комплект учебной мебели, учебно-наглядные пособия в электронном виде, компьютеры с возможностью подключения к Интернет и доступом в ЭИОС; программное обеспечение: MS Windows /Linux /Ред ОС, MS Office / OpenOffice/ LibreOffice, DrWeb ES, 7-Zip, MediaPlayer Classic, Яндекс браузер / Mozilla Firefox / Internet Explorer, AST Test	394087, Воронежская область, г. Воронеж, ул. Мичурина, д.1
3	Учебная аудитория для проведения учебных занятий: комплект учебной мебели, компьютеры с возможностью подключения к "Интернет" и обеспечением доступа в ЭИОС; программное обеспечение: MS Windows /Linux /Ред ОС, MS Office / OpenOffice/LibreOffice, DrWeb ES, 7-Zip, MediaPlayer Classic, Яндекс браузер / Mozilla Firefox / Internet Explorer, AST Test	394087, Воронежская область, г. Воронеж, ул. Мичурина, д.1
4	Помещение для самостоятельной работы: комплект учебной мебели, компьютеры с возможностью подключения к "Интернет" и обеспечением доступа в ЭИОС; программное обеспечение: MS Windows /Linux /Ред ОС, MS Office / OpenOffice/LibreOffice, DrWeb ES, 7-Zip, MediaPlayer Classic, Яндекс браузер / Mozilla Firefox / Internet Explorer, AST Test	394087, Воронежская область, г. Воронеж, ул. Мичурина, д.1, а.: 113, 115, 116, 119, 120, 122, 122а, 126, 219 (с 16.00 до 20.00)

7.2. Программное обеспечение




7.2.1. Программное обеспечение общего назначения

№	Название	Размещение
1	Операционные системы MS Windows /Linux /Ред ОС	ПК в локальной сети ВГАУ
2	Пакеты офисных приложений MS Office / OpenOffice/LibreOffice	ПК в локальной сети ВГАУ
3	Программы для просмотра файлов Adobe Reader / DjVu Reader	ПК в локальной сети ВГАУ
4	Браузеры Яндекс Браузер / Mozilla Firefox / Microsoft Edge	ПК в локальной сети ВГАУ
5	Антивирусная программа DrWeb ES	ПК в локальной сети ВГАУ
6	Программа-архиватор 7-Zip	ПК в локальной сети ВГАУ
7	Мультимедиа проигрыватель MediaPlayer Classic	ПК в локальной сети ВГАУ
8	Платформа онлайн-обучения eLearning server	ПК в локальной сети ВГАУ
9	Система компьютерного тестирования AST Test	ПК в локальной сети ВГАУ

7.2.2. Специализированное программное обеспечение

1	Среда программирования Microsoft Visual Studio (msdn)	ПК ГИС-лаборатории
---	---	--------------------

8. Междисциплинарные связи

Взаимосвязанные дисциплины		Кафедра, на которой преподается дисциплина	Подпись заведующего кафедрой
Код	Название		
Б1.О.15	Алгоритмизация и программирование	Информационного обеспечения и моделирования агроэкономических систем	
Б1.О.25	Проектный практикум	Информационного обеспечения и моделирования агроэкономических систем	
Б1.В.16	Разработка приложений для мобильных устройств	Информационного обеспечения и моделирования агроэкономических систем	
Б1.В.17	Программирование офисных приложений	Информационного обеспечения и моделирования агроэкономических систем	